ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ, ΕΡΕΥΝΑΣ
ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ

ΙΝΣΤΙΤΟΥΤΟ ΕΚΠΑΙΔΕΥΤΙΚΗΣ
ΠΟΛΙΤΙΚΗΣ

Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ

Β΄ Λυκείου

ΤΟΜΟΣ 2ος

Αθήνα 2014

ΙΝΣΤΙΤΟΥΤΟ ΤΕΧΝΟΛΟΓΙΑΣ
ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΕΚΔΟΣΕΩΝ
«ΔΙΟΦΑΝΤΟΣ»

ΙΝΣΤΙΤΟΥΤΟ ΕΚΠΑΙΔΕΥΤΙΚΗΣ ΠΟΛΙΤΙΚΗΣ
Πρόεδρος: **Σωτήριος Γκλαβάς**

ΓΡΑΦΕΙΟ ΕΡΕΥΝΑΣ ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΩΝ Β΄
Προϊστάμενος:
Παύλος Φ. Μάραντος

ΣΥΓΓΡΑΦΕΙΣ:
Δρ. **Σπυρίδων Δουκάκης**, Πληροφορικός, Μαθηματικός, PIERCE-Αμερικανικό Κολλέγιο Ελλάδος
Χρήστος Δουληγέρης, Καθηγητής Τμήματος Πληροφορικής Πανεπιστημίου Πειραιώς
Δρ. **Θεόδωρος Καρβουνίδης**, Εκπαιδευτικός ΠΕ19
Χρήστος Κοίλιας, Καθηγητής Τμήματος Μηχανικών Πληροφορικής Τ.Ε. ΤΕΙ Αθήνας

Δρ. Αθανάσιος Πέρδος, Πληροφορικός, Φυσικός, Ελληνογαλλική Σχολή Καλαμαρί

ΣΥΝΤΟΝΙΣΤΗΣ: Χρήστος Κοίλιας

ΣΥΛΛΟΓΗ - ΕΠΕΞΕΡΓΑΣΙΑ ΥΛΙΚΟΥ:
Δημήτριος Κοτσιφάκος, Εκπαιδευτικός, ΠΕ 1708

ΚΡΙΤΕΣ-ΑΞΙΟΛΟΓΗΤΕΣ:
Παναγιώτης Βαρζάκας, Μέλος ΔΕΠ (συντονιστής)
Σοφία Τζελέπη, Σχολική Σύμβουλος, ΠΕ19
Πέτρος Ματζάκος, Εκπαιδευτικός, ΠΕ19

ΦΙΛΟΛΟΓΙΚΗ ΕΠΙΜΕΛΕΙΑ:
Μαρία Κοίλια

ΕΞΩΦΥΛΛΟ: Γιώργος Σκούφος

ΣΕΛΙΔΟΠΟΙΗΣΗ: Γιώργος Σκούφος

ΑΝΑΔΟΧΟΣ: ΕΚΔΟΣΕΙΣ ΝΕΩΝ ΤΕ-ΧΝΟΛΟΓΙΩΝ

NewTech Pub.

Στουρνάρη 49Α, 106 82,
Αθήνα, Τηλ. 210-38.45.594
Fax: 210-38.08.009
E-mail:
contact@newtech-publications.gr
URL:
www.newtech-pub.com

«ΔΗΜΙΟΥΡΓΙΑ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΥΛΙΚΟΥ ΓΙΑ ΤΑ ΝΕΑ ΜΑΘΗΜΑΤΑ ΤΟΥ ΓΕΝΙΚΟΥ ΛΥΚΕΙΟΥ» της Πράξης «ΝΕΟ ΣΧΟΛΕΙΟ (ΣΧΟΛΕΙΟ 21ου αιώνα) - ΝΕΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ» ΜΕ ΚΩΔ. ΟΠΣ 295450, των Αξόνων Προτεραιότητας 1, 2 και 3 - ΟΡΙΖΟΝΤΙΑ ΠΡΑΞΗ του ΕΠΙΧΕΙΡΗΣΙΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ «ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ», που συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση - Ευρωπαϊκό Κοινωνικό Ταμείο και από Εθνικούς Πόρους (ΕΣΠΑ 2007 - 2013).

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Η αξιολόγηση, η κρίση
των προσαρμογών και
η επιστημονική επιμέλεια
του προσαρμοσμένου βιβλίου
πραγματοποιείται από τη Μονάδα
Ειδικής Αγωγής του Ινστιτούτου
Εκπαιδευτικής Πολιτικής.

Η προσαρμογή του βιβλίου
για μαθητές με μειωμένη όραση
από το ΙΤΥΕ – ΔΙΟΦΑΝΤΟΣ
πραγματοποιείται με βάση τις
προδιαγραφές που έχουν
αναπτυχθεί από ειδικούς
εμπειρογνώμονες για το ΙΕΠ.

ΠΡΟΣΑΡΜΟΓΗ ΤΟΥ ΒΙΒΛΙΟΥ
ΓΙΑ ΜΑΘΗΤΕΣ
ΜΕ ΜΕΙΩΜΕΝΗ ΟΡΑΣΗ

ΙΤΥΕ - ΔΙΟΦΑΝΤΟΣ

2.2.7 Εντολές και δομές αλγορίθμου

Στην παράγραφο αυτή δίδονται διάφορα παραδείγματα αλγορίθμων, όπου εξετάζονται τα συστατικά μέρη ενός αλγορίθμου και οι τρεις συνιστώσες του (δομή ακολουθίας, δομή επιλογής και δομή επανάληψης) ξεκινώντας από τις απλούστερες και προχωρώντας προς τις συνθετότερες. Στα περιθώρια παρουσιάζονται ορισμένα βασικά εισαγωγικά στοιχεία της χρησιμοποιούμενης ψευδογλώσσας.

Κάθε αλγόριθμος διατυπωμένος σε ψευδογλώσσα ξεκινά με τη γραμμή

Αλγόριθμος όνομα_αλγορίθμου
και τελειώνει με τη γραμμή

Τέλος όνομα_αλγορίθμου

Μεταξύ αυτών των δύο γραμμών γράφονται οι εντολές του αλγορίθμου. Οι εντολές είναι λέξεις (συνήθως ρήματα σε προστακτική) ή συμβολισμοί που προσδιορίζουν μία σαφή ενέργεια. Οι λέξεις που έχουν αυστηρά καθορισμένο νόημα στην ψευδογλώσσα καλούνται δεσμευμένες λέξεις και στο πλαίσιο του βιβλίου θα γράφονται με έντονα μπλε γράμματα.

Οι εντολές γράφονται σε ξεχωριστές γραμμές. Επεξηγηματικά σχόλια μπορούν να γράφονται οπουδήποτε στο σώμα του αλγορίθμου. Ένα σχόλιο αρχίζει με το χαρακτήρα θαυμαστικό (!) και στο πλαίσιο του βιβλίου

θα γράφεται με πλάγια γράμματα.

Στη συνέχεια επεξηγούνται οι διάφορες εντολές που μπορούν να χρησιμοποιηθούν για τη σύνταξη ενός αλγορίθμου.

Αλφάβητο

Το σύνολο των χαρακτήρων που χρησιμοποιούνται στην ψευδογλώσσα περιλαμβάνει:

- όλα τα γράμματα της ελληνικής ή αγγλικής αλφαβήτου πεζά και κεφαλαία
- τους αριθμητικούς χαρακτήρες 0-9
- τους επόμενους ειδικούς χαρακτήρες

" εισαγωγικά (διπλά)

( ) παρενθέσεις

[ ] αγκύλες

* αστερίσκος

+ συν

, κόμμα

– μείον

. τελεία

/ κάθετος

! θαυμαστικό

< μικρότερο από

= ίσον

> μεγαλύτερο από

≤ μικρότερο ή ίσο

≥ μεγαλύτερο ή ίσο

≠ διάφορο

∧ άνω βέλος

_ κάτω παύλα

κενό

- και ένα γραφικό σύμβολο το ← (αριστερό βέλος)

2.2.7.1 Εκχώρηση, Είσοδος και Έξοδος τιμών

Η γενική μορφή της εντολής εκχώρησης είναι:

Μεταβλητή ← Έκφραση

και η λειτουργία της είναι «εκτελούνται οι πράξεις στην έκφραση και η τιμή της εκχωρείται (αποδίδεται, μεταβιβάζεται) στη μεταβλητή».

Στην εντολή χρησιμοποιείται το αριστερό βέλος προκειμένου να δείχνει τη φορά της εκχώρησης. Για το σκοπό αυτό χρησιμοποιούνται διάφορα σύμβολα από τις γλώσσες προγραμματισμού. Για παράδειγμα στην Pascal και Delphi χρησιμοποιείται το :=, ενώ στην Basic και τη C το =.

Προσοχή, λοιπόν, το σύμβολο της ισότητας (=) στις γλώσσες προγραμματισμού ή το αριστερό βέλος (←) στην ψευδογλώσσα, δεν είναι σύμβολο εξίσωσης. Το σύμβολο = χρησιμοποιείται ως τελεστής σύγκρισης.

Αριστερά του συμβόλου ← υπάρχει πάντα μόνο μια μεταβλητή, ενώ δεξιά μπορεί να υπάρχει σταθερά, μεταβλητή ή έκφραση.

Σταθερές

Οι σταθερές στην ψευδογλώσσα μπορεί να είναι αριθμητικές, αλφαριθμητικές ή λογικές. Για το σχηματισμό μιας αριθμητικής σταθεράς χρησιμοποιούνται οι αριθμητικοί χαρακτήρες και πιθανά ένας από τους χαρακτήρες +, –. Επίσης, μπορεί να χρησιμοποιηθεί το κόμμα για το δεκαδικό σημείο. Π.χ. 5, 123,27, -1, 1000000 κ.λπ.

Για το σχηματισμό μιας αλφαριθμητικής σταθεράς χρησιμοποιούνται οποιοιδήποτε χαρακτήρες περικλειόμενοι σε διπλά εισαγωγικά.

Μια σταθερά μπορεί να έχει οποιοδήποτε πλήθος αριθμητικών ή αλφαριθμητικών χαρακτήρων αντίστοιχα.
Οι λογικές σταθερές είναι δύο, η Αληθής και Ψευδής.

Η εκχώρηση τιμών επιτυγχάνεται και με τις εντολές εισόδου. Η εντολή

Διάβασε λίστα_μεταβλητών

επιτρέπει την είσοδο τιμών και την εκχώρηση αυτών στις μεταβλητές που αναφέρονται στη λίστα_μεταβλητών.

Η εντολή **Διάβασε** διαφέρει από την εντολή εκχώρησης, γιατί στη δεύτερη οι τιμές των μεταβλητών προσδιορίζονται κατά τη συγγραφή του αλγορίθμου, ενώ στην πρώτη κατά την εκτέλεση του αλγορίθμου.

Για την έξοδο τιμών (αποτελεσμάτων) μπορούν να χρησιμοποιηθούν οι εντολές **Γράψε, Εμφάνισε** ή **Εκτύπωσε** με ίδια σύνταξη. Κάθε μία από αυτές τις εντολές συνοδεύεται από μια λίστα μεταβλητών ή σταθερών. Π.χ. **Γράψε** "Τιμή:", αξία.

Μεταβλητές

Για το σχηματισμό του ονόματος μιας μεταβλητής χρησιμοποιείται οποιοσδήποτε αριθμός αλφαβητικών ή αριθμητικών χαρακτήρων και ο χαρακτήρας κάτω παύλα. Ο πρώτος χαρακτήρας της μεταβλητής πρέπει να είναι αλφαβητικός και δεν μπορεί να χρησιμοποιηθεί δεσμευμένη λέξη ως όνομα μεταβλητής.
Οι μεταβλητές χαρακτηρίζονται ως αριθμητικές, αλφαριθμητικές ή λογικές ανάλογα με την τιμή που θα αποδοθεί σε αυτές. Πριν από την απόδοση κάποιας τιμής σε μια μεταβλητή (με εντολή εισόδου ή εκχώρησης) η μεταβλητή έχει απροσδιόριστη τιμή.
Οι σταθερές και οι μεταβλητές καλούνται και τελεστέοι.

2.2.7.2 Δομή ακολουθίας

Η δομή ακολουθίας χρησιμοποιείται για την αντιμετώπιση προβλημάτων στα οποία οι εντολές εκτελούνται η μία μετά την άλλη από πάνω προς τα κάτω.

Παράδειγμα 2.8. Είσοδος και έξοδος αριθμών

Να διαβαστούν δύο αριθμοί και να υπολογιστεί και να εμφανιστεί το άθροισμά τους.

Αλγόριθμος Άθροισμα
Διάβασε α, β
Σ ← α + β
Εμφάνισε Σ
Τέλος Άθροισμα

Η πρώτη ενέργεια που γίνεται είναι η εισαγωγή δεδομένων. Αυτό επιτυγχάνεται με τη χρήση της εντολής Διάβασε. Μετά την εκτέλεση της εντολής αυτής στις μεταβλητές α και β έχουν εκχωρηθεί τιμές, οπότε υπάρχει η δυνατότητα επεξεργασίας των τιμών. Εδώ απαιτείται η πρόσθεση των δύο αριθμών και η απόδοση του αθροίσματος σε μια άλλη μεταβλητή, τη Σ, που επιτυγχάνεται με την επόμενη εντολή εκχώρησης. Τελευταία ενέργεια αποτελεί η εμφάνιση (ή εκτύπωση) του αποτελέσματος.

Παράδειγμα 2.9. Υπολογισμός τελικής αξίας είδους

Να γραφεί αλγόριθμος, ο οποίος να διαβάζει την καθαρή αξία ενός είδους και το ποσοστό ΦΠΑ και να υπολογίζει και να εκτυπώνει την τελική αξία.

Αλγόριθμος Υπολογισμός
Διάβασε ΚΑ, ΠΦΠΑ
ΤΑ $\leftarrow$ ΚΑ + ΚΑ * ΠΦΠΑ / 100
Εκτύπωσε "Τελική Αξία:", ΤΑ
Τέλος Υπολογισμός

Η τελική αξία (ΤΑ) ενός είδους βρίσκεται, αν στην καθαρή αξία (ΚΑ) προστεθεί η αξία ΦΠΑ. Αυτό επιτυγχάνεται με την εντολή εκχώρησης.

Ο ΦΠΑ (Φόρος Προστιθέμενης Αξίας) είναι ένας φόρος που επιβάλλεται σε κάθε προϊόν που πωλείται ή σε κάθε παρεχόμενη υπηρεσία. Σήμερα για τα περισσότερα είδη το ποσοστό του ΦΠΑ είναι 23%, για την εστίαση και είδη πρώτης ανάγκης είναι 13% και για τα βιβλία 6,5%. Όμως στα νησιά του Αιγαίου (εκτός της Κρήτης) εφαρμόζονται μειωμένοι συντελεστές. Τα ποσοστά του ΦΠΑ αλλάζουν με κυβερνητική απόφαση.

Οι εντολές εισόδου/εξόδου μπορούν να συνδυάζονται προκειμένου να είναι πιο κατανοητή η ενέργεια που απαιτείται από το χρήστη του προγράμματος που θα υλοποιεί έναν αλγόριθμο.

Εμφάνισε "Δώστε τιμές για τα α και β"
Διάβασε α, β

Εναλλακτική είσοδος και έξοδος τιμών παρέχεται με τη χρήση των εντολών Δεδομένα και Αποτελέσματα. Η εντολή **Δεδομένα** γράφεται δεύτερη (μετά την εντολή Αλγόριθμος) και περιγράφει εντός των συμβόλων // .... // τα δεδομένα του αλγορίθμου, δηλαδή τις μεταβλητές **που έχουν ήδη κάποια τιμή**. Αντίστοιχα η εντολή Αποτελέσματα γράφεται προτελευταία και περιέχει τις μεταβλητές εξόδου. Οι επόμενοι δύο αλγόριθμοι για τις ίδιες τιμές εισόδου έχουν την ίδια τιμή εξόδου.

Αλγόριθμος Άθροισμα_1
Διάβασε α, β
Σ ← α + β
Γράψε Σ
Τέλος Άθροισμα_1

Αλγόριθμος Άθροισμα_2
Δεδομένα // α, β //
Σ ← α + β

Αποτελέσματα // Σ //
Τέλος Άθροισμα_2

Η χρήση των εντολών Δεδομένα και Αποτελέσματα γενικά προτιμάται προκειμένου ο αλγόριθμος να απαλλαγεί από τις λεπτομέρειες εισόδου/εξόδου και να επικεντρωθεί στο πρόβλημα που επιλύει (εκτός βέβαια αν το πρόβλημα είναι η εισαγωγή δεδομένων). Επίσης η χρήση τους συνιστάται στην περίπτωση που τα δεδομένα εισόδου ή/και εξόδου είναι πολυπληθή, όπως για παράδειγμα σε προβλήματα επεξεργασίας πινάκων. Τέλος η χρήση τους επιβάλλεται, στην περίπτωση που ένας αλγόριθμος καλείται από άλλον (βλ. παρ. 2.2.7.5).

Αν και η χρήση της μιας ή της άλλης μεθόδου αφήνεται γενικά στην ευχέρεια του συντάκτη του αλγορίθμου, ο μαθητής χρειάζεται να είναι προσεκτικός στη χρήση των δύο εντολών. Έτσι αν η εκφώνηση ενός θέματος λέει «Να γραφεί αλγόριθμος ο οποίος να διαβάζει ....», τότε είναι απαραίτητο να χρησιμοποιηθεί η εντολή Διάβασε. Αντίθετα αν η εκφώνηση λέει «Δίδεται ένας πίνακας Α. Να γραφεί αλγόριθμος ο οποίος ...», τότε χρειάζεται να χρησιμοποιηθεί η εντολή Δεδομένα.

Τελεστές

Τελεστές είναι τα σύμβολα και οι

λέξεις που χρησιμοποιούνται στις διάφορες πράξεις. Υπάρχουν οι επόμενοι τελεστές:

➢ Αριθμητικοί

Οι αριθμητικοί τελεστές χρησιμοποιούνται για την εκτέλεση αριθμητικών πράξεων. Είναι οι:

+	για πρόσθεση
–	για αφαίρεση
*	για πολλαπλασιασμό
/	για διαίρεση
mod	για το υπόλοιπο ακέραιας διαίρεσης
div	για το πηλίκο ακέραιας διαίρεσης
^	για ύψωση σε δύναμη

➢ Σχεσιακοί

Οι σχεσιακοί τελεστές χρησιμοποιούνται για τη σύγκριση δύο τιμών. Το αποτέλεσμα μιας σύγκρισης είναι είτε Αληθής είτε Ψευδής. Οι σχεσιακοί ή συγκριτικοί τελεστές είναι οι επόμενοι:

$<$	μικρότερο
$>$	μεγαλύτερο
$=$	ίσο
$\leq$	μικρότερο ή ίσο
$\geq$	μεγαλύτερο ή ίσο
$\neq$	διάφορο

Αντί των τριών τελευταίων τελεστών μπορεί να χρησιμοποιηθούν και οι συνδυασμοί των χαρακτήρων $<=$, $>=$ και $< >$ αντίστοιχα.

➢ Λογικοί

Οι λογικοί τελεστές υλοποιούν τις λογικές πράξεις. Το αποτέλεσμα μιας λογικής πράξης είναι Αληθής ή Ψευδής. Λογικοί τελεστές είναι:

όχι	πράξη άρνησης
και	πράξη σύζευξης
ή	πράξη διάζευξης

➢ Συναρτησιακοί τελεστές ή Συναρτήσεις

Μια συνάρτηση χρησιμοποιείται για να εκτελέσει μια προκαθορισμένη λειτουργία. Κάθε συνάρτηση έχει ένα όνομα ακολουθούμενο από ζεύγος παρενθέσεων που περικλείουν μια μεταβλητή ή μια σταθερά ή γενικότερα μια έκφραση. Στην ψευδογλώσσα μπορούν να χρησιμοποιηθούν όλες οι συνηθισμένες συναρτήσεις, όπως οι τριγωνομετρικές **ΗΜ**(x), **ΣΥΝ**(x), **ΕΦ**(x), οι μαθηματικές **Α_Τ**(x) για την

απόλυτη τιμή, Ε(x) για την e^x, ΛΟΓ(x) για το δεκαδικό λογάριθμο, ΛΝ(x) για το φυσικό λογάριθμο, Τ_Ρ(x) για την τετραγωνική ρίζα, και Α_Μ(x) για το ακέραιο μέρος.

2.2.7.3 Δομή επιλογής

Στην πράξη πολύ λίγα προβλήματα μπορούν να επιλυθούν με τον προηγούμενο τρόπο της σειριακής / ακολουθιακής δομής ενεργειών. Συνήθως τα προβλήματα έχουν κάποιες ιδιαιτερότητες και δεν μπορούν να εκτελεστούν τα ίδια βήματα για κάθε περίπτωση. Τις πιο πολλές φορές λαμβάνονται κάποιες αποφάσεις με βάση κάποια κριτήρια που μπορεί να είναι διαφορετικά για κάθε στιγμιότυπο ενός προβλήματος. Για παράδειγμα

το πρόβλημα της εξόδου (από το σπίτι) σχετίζεται με τις καιρικές συνθήκες. Έτσι κάποιος μπορεί να πει ότι, «αν βρέχει, θα πάρω ομπρέλα».

Με τη δομή επιλογής μπορεί να τροποποιηθεί η σειρά εκτέλεσης των εντολών ενός αλγορίθμου. Η διαδικασία επιλογής περιλαμβάνει τον έλεγχο μιας συνθήκης που μπορεί να έχει δύο τιμές (Αληθής ή Ψευδής) και ακολουθεί η απόφαση εκτέλεσης εντολών με βάση την τιμή αυτής της συνθήκης. Ως συνθήκη εννοείται μια λογική έκφραση στην οποία υπάρχει τουλάχιστον ένας σχεσιακός τελεστής (δηλαδή η συνθήκη δεν μπορεί να απαρτίζεται από μόνο μια μεταβλητή ή μια σταθερά ή μια αριθμητική παράσταση).

Εκφράσεις

Μια έκφραση μπορεί να είναι μια σταθερά, μια μεταβλητή, μια συνάρτηση ή ένας συνδυασμός σταθερών, μεταβλητών, συναρτήσεων, τελεστών και παρενθέσεων.

Σε μία έκφραση που αποτελείται από συνδυασμό στοιχείων, εκτελούνται οι πράξεις επί των σταθερών και μεταβλητών που ορίζουν οι τελεστές. Οι χρησιμοποιούμενοι τελεστές έχουν διαφορετική ιεραρχία. Αυτό σημαίνει ότι κάποιες πράξεις μπορεί να προηγούνται από κάποιες άλλες σε μια έκφραση.

Η ιεραρχία των πράξεων είναι η ακόλουθη:

Α. Αριθμητικοί τελεστές

Σε κάθε έκφραση που υπάρχουν αριθμητικοί τελεστές, ακολουθείται η προσδιορισμένη από τα μαθηματικά ιεραρχία των πράξεων.

1. Ύψωση σε δύναμη
2. Πολλαπλασιασμός, Διαίρεση, Πηλίκο ακέραιας διαίρεσης, Υπόλοιπο ακέραιας διαίρεσης
3. Πρόσθεση, Αφαίρεση

Β. Σχεσιακοί τελεστές

Γ. Λογικοί τελεστές

1. όχι
2. και
3. ή

Αν οι πράξεις είναι ίδιας ιεραρχίας, τότε εκτελούνται από τα αριστερά προς τα δεξιά.
Οι τελεστέοι ενός αριθμητικού τελεστή πρέπει να είναι αριθμητικές εκφράσεις. (π.χ. α + β^3)
Στις λογικές εκφράσεις μπορούν να χρησιμοποιηθούν όλοι οι τελεστές. Αν μία λογική έκφραση περιλαμβάνει τελεστές, τότε ένας τουλάχιστον πρέπει να είναι λογικός ή συγκριτικός. (π.χ. α + β^3 > > 5 και γ / 3 < 2)
Οι συγκριτικοί τελεστές

συνδυάζονται με εκφράσεις ιδίου τύπου, ενώ οι λογικοί τελεστές μόνο με λογικές εκφράσεις. (π.χ. $α + β > 3$, "ΑΒ" < "Γ")

Οι συγκρίσεις λογικών εκφράσεων έχουν νόημα μόνο στην περίπτωση του = και ≠.

➢ **Αριθμητικές εκφράσεις**

Στη συνέχεια παρουσιάζονται μερικά παραδείγματα και διευκρινίσεις που αφορούν τις αριθμητικές εκφράσεις.

❑ Στην έκφραση 5 + 12 / 3 * 2 – – 1 οι πράξεις εκτελούνται με την επόμενη σειρά

1. 12 / 3 (= 4)
2. 4 * 2 (= 8)
3. 5 + 8 (= 13)
4. 13 – 1 (= 12)

Σε μια έκφραση μπορούν να χρησιμοποιηθούν και παρενθέσεις. Οι παρενθέσεις μπορεί να μεταβάλλουν την προτεραιότητα των πράξεων.

❑ **Στην έκφραση 4 * (1 + 2) εκτελείται πρώτα η πρόσθεση (1 + 2 = = 3) και μετά ο πολλαπλασιασμός (4 * 3 = 12)**

❑ **Παρατίθεται ένας πίνακας με τη γραφή μερικών μαθηματικών τύπων ως εκφράσεις της ψευδογλώσσας.**

Πίνακας 2.4. Μαθηματικοί τύποι ως εκφράσεις της ψευδογλώσσας

Μαθηματικός τύπος	Έκφραση ψευδογλώσσας
$\dfrac{x - y}{z}$	(x – y) / z
$\dfrac{xy}{z + 1}$	x * y / (z + 1)
$(x^2)^3$	(x^2)^3
x^{2^3}	x^(2^3)
$x(-y)$	x * (–y)

Στην ακέραια διαίρεση οι τελεστέοι των τελεστών div και mod είναι υποχρεωτικά θετικοί ακέραιοι αριθμοί.

➢ **Λογικές εκφράσεις**

Οι σχεσιακοί τελεστές χρησιμοποιούνται για τη σύγκριση δύο τιμών. Το αποτέλεσμα μιας σύγκρισης μπορεί να είναι είτε Αληθής είτε Ψευδής.

❑ Στην έκφραση $x + y < (z - 1) / t$ το αποτέλεσμα είναι Αληθής, αν το αποτέλεσμα της πράξης $x + y$ είναι μικρότερο από το αποτέλεσμα της πράξης $z - 1$ διαιρούμενο δια t.

Οι σχεσιακοί τελεστές μπορούν να χρησιμοποιηθούν και με αλφαριθμητικούς τελεστέους. Η σύγκριση αλφαριθμητικών εκφράσεων πραγματοποιείται βάσει διεθνών προτύπων που στοχεύουν στην κωδικοποίηση όλων των συστημάτων γραφής. Ο υπολογισμός του αποτελέσματος, λοιπόν, της σύγκρισης αλφαριθμητικών εκφράσεων που περιέχουν οποιονδήποτε χαρακτήρα είναι πέρα από τους στόχους του μαθήματος. Για το λόγο αυτό, στην ψευδογλώσσα θα γίνεται σύγκριση αλφαριθμητικών εκφράσεων που περιέχουν μόνο τα κεφαλαία γράμματα του ελληνικού αλφαβήτου, στα οποία ισχύει

η αλφαβητική σειρά. Για παράδειγμα η λογική έκφραση "ΑΔΓ" > "ΑΒΚ" είναι αληθής, διότι κατά τη σύγκριση χαρακτήρα προς χαρακτήρα από αριστερά προς τα δεξιά εντοπίζεται ότι το γράμμα Δ είναι διαφορετικό και μεγαλύτερο του γράμματος Β.

Οι λογικοί τελεστές πραγματοποιούν τις λογικές πράξεις σε μια έκφραση. Το αποτέλεσμα μιας λογικής πράξης είναι πάντα Αληθής ή Ψευδής, σύμφωνα με τον επόμενο πίνακα τιμών, όπου με Χ και Υ εννοούνται δύο λογικές εκφράσεις, στις οποίες χρησιμοποιούνται μόνο αριθμητικοί και σχεσιακοί τελεστές.

Πίνακας 2.5. Πίνακας τιμών δύο λογικών εκφράσεων (όχι, και, ή)

X	Y	όχι X	X και Y	X ή Y
Αληθής	Αληθής	Ψευδής	Αληθής	Αληθής
Αληθής	Ψευδής	Ψευδής	Ψευδής	Αληθής
Ψευδής	Αληθής	Αληθής	Ψευδής	Αληθής
Ψευδής	Ψευδής	Αληθής	Ψευδής	Ψευδής

Σε μια λογική έκφραση οι λογικές πράξεις εκτελούνται μετά τις αριθμητικές και συγκριτικές.

❑ **Η σχέση $0 < x < 2k + 1$ γράφεται στην ψευδογλώσσα $x > 0$ και $x < 2 * k + 1$ και είναι αληθής, αν x θετικό και ταυτόχρονα μικρότερο του $2 * k + 1$.**

Ο τελεστής όχι έχει έναν τελεστέο, ενώ οι και, ή έχουν δύο (ή περισσότερους).

Προσοχή. Ο συγκριτικός τελεστής ≤ είναι ένας και διαβάζεται «μικρότερο ή ίσο». Δεν πρέπει να αναλύεται και σε μια λογική έκφραση με χρήση του λογικού τελεστή **ή**, διότι υπάρχει μεγάλη πιθανότητα να προκληθεί λάθος, ιδιαίτερα αν υπάρχουν πολλοί τελεστές σε μια σύνθετη έκφραση.

❑ **Η σχέση 0 < x ≤ 10 γράφεται στην ψευδογλώσσα x > 0 και x ≤ 10**

Στην πιο πάνω έκφραση δεν μπορούμε να αναλύσουμε την x ≤ 10 σε x = 10 ή x < 10, διότι θα λάβουμε την x > 0 και x = 10 ή x < 10 η οποία για x = –1 θα δώσει τιμή Αληθής.
Συνιστάται ανεπιφύλακτα να χρησιμοποιούνται παρενθέσεις όταν σε έκφραση υπάρχουν πολλοί λογικοί τελεστές.

Παράδειγμα 2.10. Να διαβαστεί ένας αριθμός και να εμφανιστεί η απόλυτη τιμή του.

Η απόλυτη τιμή ενός αριθμού είναι ο ίδιος ο αριθμός, αν είναι θετικός ή ο αντίθετός του, αν είναι αρνητικός. Έτσι για να υπολογιστεί η απόλυτη τιμή ενός αριθμού αρκεί να ελεγχθεί, αν τυχόν ο δεδομένος αριθμός είναι αρνητικός και αν ναι, να βρεθεί ο αντίθετός του. Ο συλλογισμός αυτός οδηγεί στον επόμενο αλγόριθμο.

```
Αλγόριθμος Απόλυτη_τιμή1
Διάβασε α
Αν α < 0 τότε
        α ← α * (–1)
        ! α ← – α
Τέλος_αν
```

Εμφάνισε α
Τέλος Απόλυτη_τιμή1

Αλγόριθμος Απόλυτη_τιμή2
Διάβασε α
! Η εμφάνιση της απόλυτης τιμής
! μπορεί να γίνει με τη χρήση της
! συνάρτησης Α_Τ(α)
Εμφάνισε Α_Τ(α)
Τέλος Απόλυτη_τιμή2

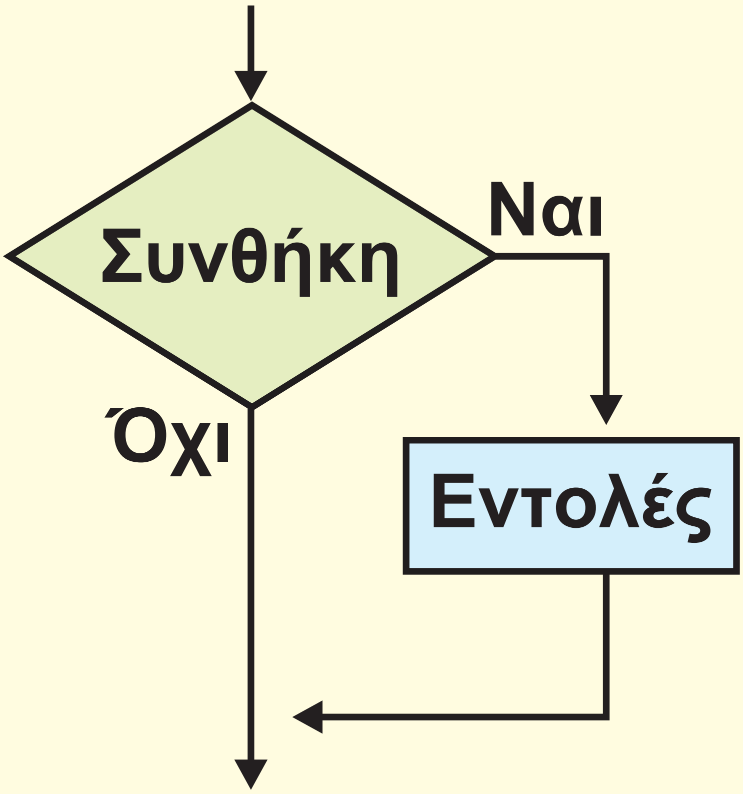
Σύγκριση αριθμών με απλή επιλογή

Απλή εντολή επιλογής

Αν Συνθήκη **τότε**
Εντολές
Τέλος_αν

Αν η συνθήκη είναι αληθής, τότε εκτελούνται οι εντολές. Οι εντολές

μπορούν να είναι μία ή περισσότε-
ρες.



Εικόνα 2.20. Διάγραμμα ροής της απλής εντολής επιλογής

Παράδειγμα 2.11. Να αναπτύξετε αλγόριθμο ο οποίος με δεδομένα τα μήκη τριών ευθυγράμμων τμημάτων θα υπολογίζει και θα εμφανίζει το εμβαδόν του τριγώνου που μπορούν να σχηματίσουν, με βάση τον τύπο του Ήρωνα $E = \sqrt{τ(τ - α)(τ - β)(τ - γ)}$, όπου τ είναι η ημιπερίμετρος του τριγώνου $τ = (α + β + γ) / 2$ και α, β, γ τα μήκη των ευθυγράμμων τμημάτων. Σε περίπτωση που τα ευθύγραμμα τμήματα δεν μπορούν να σχηματίσουν τρίγωνο, εμφανίζεται κατάλληλο μήνυμα. Για να σχηματιστεί τρίγωνο θα πρέπει το άθροισμα των μηκών δύο οποιονδήποτε ευθυγράμμων τμημάτων να είναι μεγαλύτερο από το μήκος του άλλου τμήματος.

Αλγόριθμος Εμβαδό
Δεδομένα // α, β, γ //
Αν $α + β > γ$ **και** $β + γ > α$ **και** $γ + α > β$
τότε
    $τ \leftarrow (α + β + γ) / 2$

Εμβ ← **T_P**(τ * (τ – α) * (τ – β) * * (τ – γ))
Εμφάνισε Εμβ
αλλιώς
Εμφάνισε "Δεν σχηματίζεται τρίγωνο"
Τέλος_αν
Τέλος Εμβαδό

Παράδειγμα 2.12. Το όζον (O_3) αποτελεί έναν από τους ρύπους που προκαλούν μόλυνση στην ατμόσφαιρα. Σε περιπτώσεις που ο ρύπος αυτός ξεπεράσει τα 300 μg/m^3 τότε πρέπει να ληφθούν μέτρα. Να αναπτυχθεί αλγόριθμος ο οποίος θα διαβάζει την τιμή του O_3 και θα εκτυπώνει το αντίστοιχο μήνυμα σύμφωνα με τον παρακάτω πίνακα:

Τιμές O_3 ($\mu g/m^3$)	Μήνυμα
Τιμή > 250	Προειδοποίηση
Τιμή > 300	Μέτρα Α
Τιμή > 500	Μέτρα Β

Επιπλέον, σε περίπτωση που έχουν ξεπεραστεί τα όρια, θα εκτυπώνει κατά πόσο τα ξεπέρασε.

Αλγόριθμος Όζον1
Διάβασε τ
Αν τ > 250 **και** τ ≤ 300 **τότε**
    Εκτύπωσε "Προειδοποίηση"
αλλιώς_αν τ > 300 **και** τ ≤ 500 **τότε**
    πο ← τ – 300
    Εκτύπωσε "Μέτρα Α", πο

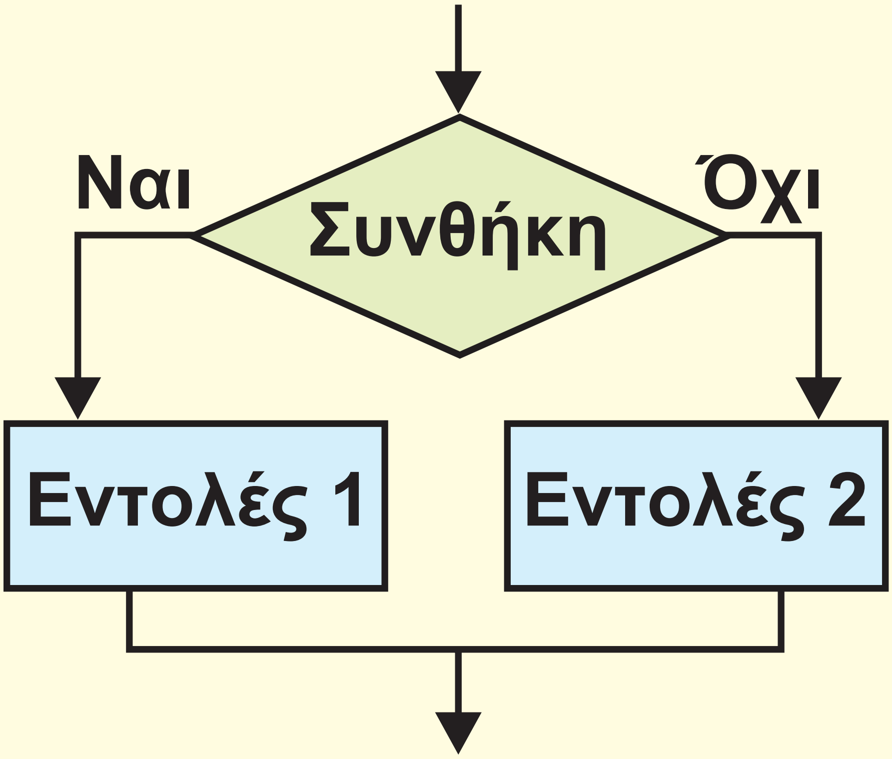
```
αλλιώς_αν τ > 500 τότε
    πο ← τ – 300
    Εκτύπωσε  "Μέτρα Β", πο
Τέλος_αν
Τέλος Όζον1
```

```
Αλγόριθμος Όζον2
Διάβασε τ
Αν τ > 500 τότε
    πο ← τ – 300
    Εκτύπωσε  "Μέτρα Β", πο
αλλιώς_αν τ > 300 τότε
    πο ← τ – 300
    Εκτύπωσε  "Μέτρα Α", πο
αλλιώς_αν τ > 250 τότε
    Εκτύπωσε  "Προειδοποίηση"
Τέλος_αν
Τέλος Όζον2
```

Σύνθετη εντολή επιλογής

```
Αν Συνθήκη τότε
            Εντολές_1
αλλιώς
            Εντολές_2
Τέλος_αν
```

Αν η συνθήκη είναι αληθής, τότε εκτελούνται οι εντολές 1, αλλιώς (δηλαδή αν η συνθήκη είναι ψευδής) εκτελούνται οι εντολές 2.



Εικόνα 2.21.
Διάγραμμα ροής της σύνθετης εντολής επιλογής

Εμφωλευμένες εντολές επιλογής

Σε όλες τις προηγούμενες περιπτώσεις όπου αναφέρεται εντολή ή εντολές, τίποτα δεν απαγορεύει αυτές οι εντολές να είναι επίσης εντολές επιλογής. Αναφερόμαστε τότε σε εμφωλευμένες εντολές επιλογής.

Παράδειγμα 2.13. Αριθμομηχανή

Να αναπτυχθεί αλγόριθμος, ο οποίος:

1. Θα διαβάζει πρώτα έναν αριθμό α, στη συνέχεια έναν από τους χαρακτήρες +, –, *, /, ανάλογα με την πράξη που θα εκτελέσει και τέλος έναν αριθμό β.
2. Θα εκτελεί την αντίστοιχη πράξη και θα τυπώνει το αποτέλεσμα. Σε

περίπτωση που έχει επιλεγεί η πράξη της διαίρεσης, ο αλγόριθμος πρέπει να ελέγχει αν το β είναι μηδέν και τότε να τυπώνει το μήνυμα «Προσοχή, διαίρεση με το μηδέν» και να οδηγείται στο τέλος του.

3. **Θα εκτυπώνει το μήνυμα «Λάθος πράξη», αν για το χαρακτήρα της πράξης δοθεί άλλο σύμβολο.**

Αλγόριθμος Αριθμομηχανή
Διάβασε α, πράξη, β
Αν πράξη = "+" **τότε**
        Εμφάνισε α + β
αλλιώς_αν πράξη = "–" **τότε**
        Εμφάνισε α – β
αλλιώς_αν πράξη = "*" **τότε**
        Εμφάνισε α * β
αλλιώς_αν πράξη = "/ " **τότε**

```
        Αν β ≠ 0 τότε
                Εμφάνισε α / β
        αλλιώς
                Εμφάνισε "Προσοχή,
                διαίρεση με το μηδέν"
        Τέλος_αν
αλλιώς
        Εμφάνισε "Λάθος πράξη"
Τέλος_αν
Τέλος Αριθμομηχανή
```

Πολλαπλή

```
Αν συνθήκη_1 τότε
        εντολές_1
αλλιώς_αν συνθήκη_2 τότε
        εντολές_2
..................
αλλιώς_αν συνθήκη_ν τότε
        εντολές_ν
```

```
αλλιώς
        εντολές_αλλιώς
Τέλος_αν
```

Αν η συνθήκη_k είναι αληθής, εκτελούνται οι εντολές_k και η συνέχεια είναι η επόμενη εντολή από το Τέλος_αν. Εφόσον καμία συνθήκη δεν είναι αληθής, τότε εκτελούνται οι εντολές_αλλιώς. Οι εντολές_αλλιώς χρησιμοποιούνται κατά περίσταση.

Σημειώνεται ότι, αν σε μια πολλαπλή εντολή επιλογής υπάρχουν πάνω από μία συνθήκες αληθείς, τότε εκτελούνται οι εντολές που ανήκουν στην πρώτη αληθή συνθήκη κατά σειρά.

2.2.7.4 Δομή επανάληψης

Λίγοι αλγόριθμοι χρησιμοποιούν μόνο τις δομές ακολουθίας και επιλογής. Στα ρεαλιστικά προβλήματα χρειάζεται συνήθως μια σειρά εντολών να επαναληφθεί πολλές φορές. Άλλωστε σε τέτοια προβλήματα «αξίζει τον κόπο» να εκπονηθεί κάποιος αλγόριθμος και στη συνέχεια να υλοποιηθεί ένα αντίστοιχο πρόγραμμα υπολογιστή.

Οι επαναληπτικές διαδικασίες μπορεί να έχουν διάφορες μορφές και να εμπεριέχουν συνθήκες επιλογών, όπως αυτές που περιγράφηκαν στις προηγούμενες παραγράφους.

Παράδειγμα 2.14. Να εκπονηθεί αλγόριθμος ο οποίος με δεδομένο ένα θετικό ακέραιο αριθμό θα εμφανίζει τους ακέραιους αριθμούς από το 1 μέχρι και τον δεδομένο αριθμό Ν.

Οι ζητούμενοι αριθμοί μπορούν να παραχθούν με ένα συστηματικό τρόπο, αφού ο καθένας δημιουργείται από τον προηγούμενό του προσθέτοντας το 1. Με την αξιοποίηση αυτού του γεγονότος, ο αλγόριθμος δημιουργεί κάθε νέο αριθμό σε μια μεταβλητή, έστω i. Αυτό μπορεί να γίνει με τη χρήση της εντολής εκχώρησης: $i \leftarrow i + 1$.

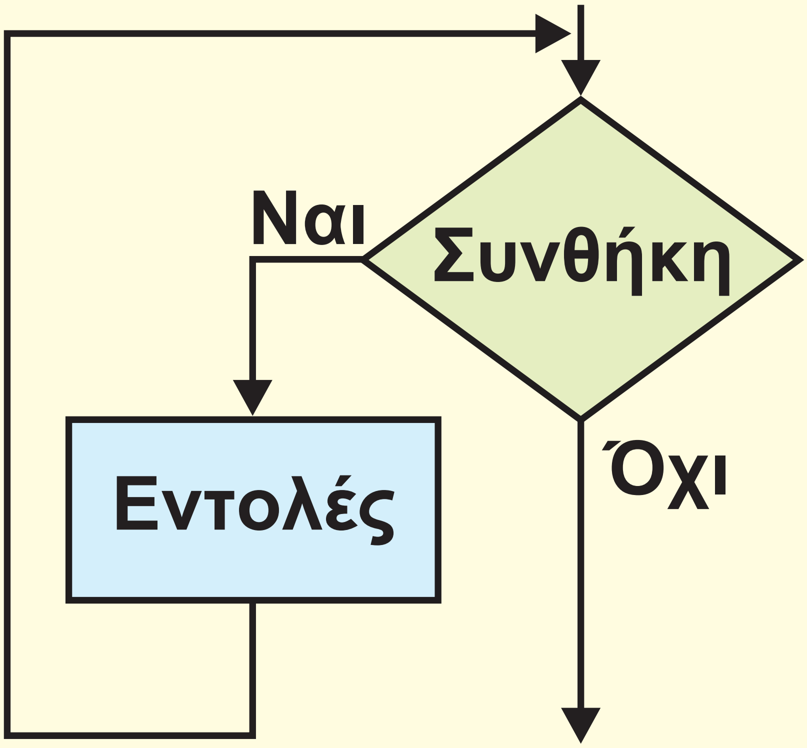
Οπότε προκύπτει το ακόλουθο:

```
i ← 1
Εμφάνισε i     ! Εμφανίζεται το 1
i ← i + 1
Εμφάνισε i     ! Εμφανίζεται το 2
i ← i + 1
Εμφάνισε i     ! Εμφανίζεται το 3
.....
```

Εντολή
Όσο ... επανάλαβε

```
Όσο Συνθήκη επανάλαβε
            Εντολές
Τέλος_επανάληψης
```

Εκτελούνται οι εντολές όσο η συνθήκη είναι αληθής



Εικόνα 2.22. Διάγραμμα ροής της Όσο...επανάλαβε

Το ζεύγος των εντολών i ← i + 1 και Εμφάνισε i επαναλαμβάνεται αυτούσιο. Αν μπορούσαν οι εντολές αυτές να γραφούν μία φορά και να εκτελεστούν Ν φορές, τότε το πρόβλημα θα λυνόταν. Αυτό επιτυγχάνεται με τις εντολές επανάληψης. Το πόσες

φορές μπορούν να εκτελεστούν οι εντολές επανάληψης καθορίζεται με διαφορετικούς τρόπους. Στο παράδειγμα οι εντολές εκτελούνται όσο διάστημα η μεταβλητή i είναι μικρότερη ή ίση της μεταβλητής Ν.

Η εντολή i ← i + 1 δεν είναι εξίσωση (γιατί και να ήταν δεν έχει λύση), αλλά δρα ως εξής: κάθε φορά που εκτελείται, το περιεχόμενο της μεταβλητής i αυξάνεται κατά 1.

Κατόπιν αυτών ο αλγόριθμος γίνεται

Αλγόριθμος Σειρά_αριθμών
Δεδομένα // Ν //
i ← 1

```
Όσο i ≤ N επανάλαβε
          Εμφάνισε i
          i ← i + 1
Τέλος_επανάληψης
Τέλος Σειρά_αριθμών
```

Συχνά η μεταβλητή i αποκαλείται μετρητής, επειδή αυξάνεται κατά 1. Σε άλλες περιπτώσεις όμως το βήμα αύξησης μπορεί να είναι οποιοδήποτε.

Όλες οι εντολές επανάληψης μπορούν να πραγματοποιούν την εκτέλεση ενός συνόλου εντολών πάνω από μια φορά.

Στην εντολή Όσο... επανάλαβε οι εμπεριεχόμενες εντολές μπορεί να μην εκτελεστούν ποτέ, αφού η

συνθήκη τερματισμού ελέγχεται στην αρχή.

Παράδειγμα 2.15. Να γραφεί αλγόριθμος ο οποίος διαβάζει το όνομα ενός μαθητή, τους βαθμούς του σε τρία μαθήματα και υπολογίζει και τυπώνει το μέσο όρο του. Ο αλγόριθμος να σταματάει, όταν για όνομα μαθητή δοθεί το κενό εμφανίζοντας το πλήθος των μαθητών για τους οποίους υπολογίστηκε ο μέσος όρος.

Αλγόριθμος Μέσος_όρος
π ← 0
Διάβασε όνομα
Όσο όνομα ≠ " " **επανάλαβε**
    Διάβασε α, β, γ

```
      Εμφάνισε (α + β + γ) / 3
      π ← π + 1
      Διάβασε όνομα
Τέλος_επανάληψης
Εμφάνισε π
Τέλος Μέσος_όρος
```

Οι εντολές που συγκροτούν μια εντολή επανάληψης αποκαλούνται **βρόχος** (αγγλ. loop, γαλ. boucle).

Προσοχή: βρόχος, όχι βρόγχος. Βρόχος = θηλιά, βρόγχος = = πνευμόνι.

Παράδειγμα 2.16. Σε ένα σουπερμάρκετ κάθε πελάτης δικαιούται μια δωροεπιταγή 6 € αν συμπληρώσει 200 πόντους. Να αναπτυχθεί αλγόριθμος ο οποίος θα διαβάζει τους πόντους που κερδίζει ένας συγκεκριμένος πελάτης σε κάθε επίσκεψη στο σουπερμάρκετ και θα εμφανίζει μετά από πόσες επισκέψεις παίρνει τη δωροεπιταγή και ποιος είναι ο μέσος όρος πόντων σε κάθε επίσκεψη.

```
Αλγόριθμος Παράδειγμα
Σ ← 0
π ← 0
Όσο Σ < 200 επανάλαβε
        Διάβασε πόντοι
        Σ ← Σ + πόντοι
        π ← π + 1
Τέλος_επανάληψης
ΜΟ ← Σ / π
```

Εμφάνισε π, ΜΟ
Τέλος Παράδειγμα

Παράδειγμα 2.17. Κατάλογος επιλογών

Πολύ συχνά στις εφαρμογές προβάλλεται στην οθόνη ένας κατάλογος από δυνατές επιλογές (menu) και στη συνέχεια ζητείται από το χρήστη να διαλέξει μία μόνο από αυτές. Στην πιο απλή περίπτωση, οι δυνατές επιλογές είναι αριθμημένες, οπότε απλά ζητείται η εισαγωγή ενός ακέραιου αριθμού.

...
Επανάλαβε
    Εμφάνισε "1. Ενημέρωση"
    Εμφάνισε "2. Εκτύπωση"
    Εμφάνισε "3. Έξοδος"

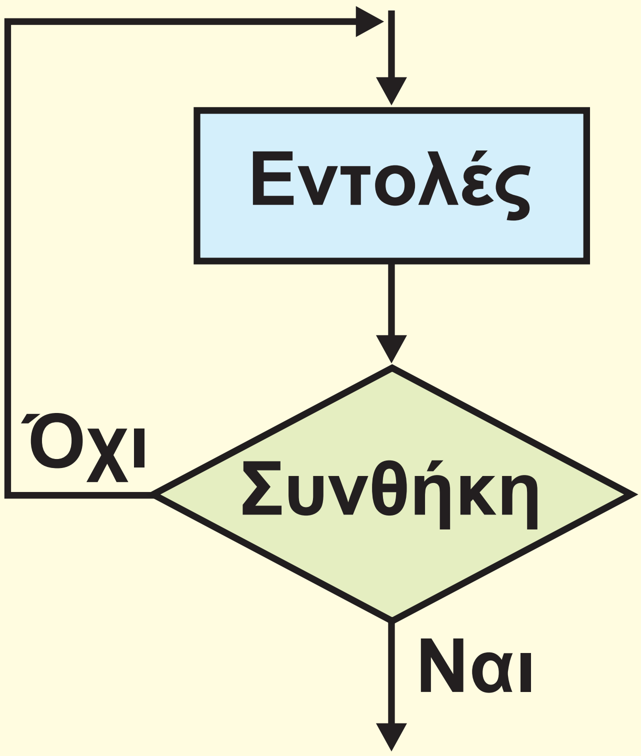
Εμφάνισε "Επιλογή:"
Διάβασε Επιλογή
Μέχρις_ότου Επιλογή = 1 **ή** Επιλογή = 2 **ή** Επιλογή = 3

Εντολή

Επανάλαβε ... Μέχρις_ότου

Επανάλαβε
Εντολές
Μέχρις_ότου Συνθήκη

Εκτελούνται οι εντολές μέχρις ότου η συνθήκη γίνει αληθής.



Εικόνα 2.23. Διάγραμμα ροής της Επανάλαβε...Μέχρις_ότου

Στο παραπάνω τμήμα αλγορίθμου, ο βρόχος επαναλαμβάνεται μέχρι να δοθεί 1, 2 ή 3. Με τον τρόπο αυτό προστατεύεται το πρόγραμμα εφαρμογής από τυχόν λανθασμένη εισαγωγή τιμών από τον χρήστη. Ας σημειωθεί με την ευκαιρία, ότι τα προγράμματα εισαγωγής δεδομένων είναι συνήθως τα πιο δύσκολα και μακροσκελή λόγω της ανάγκης να είναι φιλικά προς το χρήστη.

Υποτίθεται ότι ανάλογα με την επιλογή, ο πιο πάνω αλγόριθμος διακλαδίζεται σε άλλα σημεία, όπου υπάρχουν οι εντολές υλοποίησης κάθε λειτουργίας.

Παράδειγμα 2.18. Να εκπονηθεί αλγόριθμος ο οποίος θα διαβάζει 100 αριθμούς και θα υπολογίζει και θα εμφανίζει το άθροισμά τους.

Αλγόριθμος Άθροισμα_Αριθμών
Σ ← 0
Για i **από** 1 **μέχρι** 100
        Διάβασε α

Σ ← Σ + α
Τέλος_επανάληψης
Εμφάνισε "Άθροισμα:", Σ
Τέλος Άθροισμα_Αριθμών

Σε αυτό το παράδειγμα η εντολή **Για...από...μέχρι** περιλαμβάνει όλα τα απαραίτητα δεδομένα για την επανάληψη, δηλαδή αρχική τιμή της μεταβλητής i, τελική τιμή και το βήμα μεταβολής που είναι 1 και παραλείπεται. Ο βρόχος εκτελείται για όλες τις τιμές της μεταβλητής i.

Η εντολή εκχώρησης Σ ← Σ + α δεν είναι εξίσωση και καλύτερα να διαβάζεται ως «η νέα τιμή της μεταβλητής Σ είναι η παλιά συν α».

Συχνά η μεταβλητή Σ αποκαλείται αθροιστής, γιατί της εκχωρείται το τρέχον και τελικό άθροισμα των αριθμών και δεν πρέπει να λησμονείται ότι απαιτείται ο μηδενισμός του πριν από την έναρξη της επαναληπτικής διαδικασίας.

Η χρήση της εντολής **Για...από... μέχρι** γενικά προτιμάται όταν είναι γνωστός ο αριθμός των φορών που θα γίνει μια επανάληψη, με άλλα λόγια όταν είναι γνωστά τα τ1, τ2 και β.

Εντολή Για ... από ... μέχρι

Για μεταβλητή **από** τ1 **μέχρι** τ2 [**με_βήμα** β]
Εντολές
Τέλος_επανάληψης

Εκτελούνται οι εντολές με αρχική τιμή της μεταβλητής τ1 μέχρι και την τελική τιμή της μεταβλητής τ2.

Στη δομή αυτή τ1, τ2 είναι αριθμητικές σταθερές, μεταβλητές ή εκφράσεις. Πρέπει τ1 ≤ τ2, αν β > 0 και τ1 ≥ τ2, αν β < 0. Το βήμα β, αν είναι 1, παραλείπεται. Οι τιμές των τ1, τ2 και β μπορεί να είναι ακέραιες ή πραγματικές.

Αν τ1 > τ2 και β = 0 δεν θα εκτελεστούν οι εμπεριεχόμενες εντολές της Για, ενώ αν τ1 ≤ τ2 και β = 0 η εντολή επανάληψης θα εκτελείται άπειρες φορές (ατέρμονας βρόχος).

Εμφωλευμένες εντολές επανάληψης

Σε όλες τις προηγούμενες περιπτώσεις όπου αναφέρεται εντολή ή εντολές, τίποτα δεν απαγορεύει αυτές οι εντολές να είναι επίσης εντολές επανάληψης. Αναφερόμαστε τότε σε εμφωλευμένες εντολές επανάληψης.

Γενικά οι εμφωλευμένες εντολές επιτρέπουν το συνδυασμό συνιστωσών ενός αλγορίθμου με οποιαδήποτε σειρά. Για παράδειγμα: Επανάληψη μέσα σε επιλογή, επανάληψη μέσα σε επανάληψη κ.α.

2.2.7.5 Κλήση αλγόριθμου από αλγόριθμο

Ένας αλγόριθμος μπορεί να κληθεί από έναν άλλο αλγόριθμο με χρήση της εντολής Κάλεσε.

Η εντολή **Κάλεσε** συνοδεύεται με το όνομα του καλούμενου αλγορίθμου ακολουθούμενο πιθανά από λίστα μεταβλητών ή σταθερών μέσα σε παρενθέσεις. Οι τιμές των μεταβλητών ή σταθερών μεταβιβάζονται κατά την κλήση στις αντίστοιχες μεταβλητές της γραμμής Δεδομένα του καλούμενου αλγορίθμου. Όταν ο καλούμενος αλγόριθμος τερματίσει τη λειτουργία του, γίνεται επιστροφή στην αμέσως επόμενη εντολή της Κάλεσε. Κατά την επιστροφή μπορεί

επίσης να μεταβιβάζονται τιμές της εντολής Αποτελέσματα.

Η επικοινωνία μεταξύ των δύο αλ-
γορίθμων γίνεται ως εξής:

Παράδειγμα 2.19.

Αλγόριθμος Καλών
Διάβασε α, β
Κάλεσε Καλούμενος (α, β, γ, δ)
Γράψε "Άθροισμα:", γ
Γράψε "Διαφορά:", δ
Τέλος Καλών

Αλγόριθμος Καλούμενος
Δεδομένα // x, y //
z1 ← x + y
z2 ← x – y
Αποτελέσματα // z1, z2 //
Τέλος Καλούμενος

Η αντιστοιχία των μεταβλητών των δύο αλγορίθμων γίνεται με τη σειρά που αναφέρονται στις αντίστοιχες γραμμές, δηλ. α ⇒ x, β ⇒ y και γ ⇐ z1, δ ⇐ z2. Συνώνυμες μεταβλητές διαφορετικών αλγορίθμων δεν έχουν καμία σχέση μεταξύ τους.

Παράδειγμα 2.20. Υπολογισμός συνδυασμών

Να εκπονηθεί αλγόριθμος ο οποίος να υπολογίζει το πλήθος των συνδυασμών των Ν πραγμάτων ανά Κ. Ο συνδυασμός των Ν πραγμάτων ανά Κ συμβολίζεται με $\binom{n}{k}$ και δίνεται από τον τύπο $\frac{n!}{k!(n-k)!}$.

Αλγόριθμος Παραγοντικό
Δεδομένα // n //
nfact ← 1
Για i **από** 2 **μέχρι** n
        nfact ← nfact * i
Τέλος_επανάληψης
Αποτελέσματα // nfact //
Τέλος Παραγοντικό

Αλγόριθμος Συνδυασμοί
Δεδομένα //n, k//
Κάλεσε Παραγοντικό(n, nfact)
Κάλεσε Παραγοντικό(k, kfact)
Κάλεσε Παραγοντικό(n – k, nkfact)
combin ← nfact / (kfact * nkfact)
Αποτελέσματα //combin//
Τέλος Συνδυασμοί

Από τον τύπο των συνδυασμών φαίνεται ότι απαιτείται ο υπολογισμός του παραγοντικού τριών διαφορετικών ποσοτήτων. Για το σκοπό αυτό δημιουργείται ο αλγόριθμος Παραγοντικό, που υπολογίζει το παραγοντικό ενός αριθμού. Σε αυτόν τον αλγόριθμο με μια απλή επανάληψη βρίσκεται το παραγοντικό της μεταβλητής n στη μεταβλητή nfact. Ας προσεχθεί εδώ, ότι

επειδή ζητείται γινόμενο, η αρχική τιμή του nfact είναι 1.

Οι συνδυασμοί απαντούν στο ερώτημα κατά πόσους διαφορετικούς τρόπους είναι δυνατό π.χ. να εξαχθούν 5 φύλλα από μια τράπουλα των 52 φύλλων.

Ο αλγόριθμος Παραγοντικό καλείται τρεις φορές από τον αλγόριθμο Συνδυασμοί. Σε κάθε κλήση μεταβιβάζεται η τιμή του αριθμού για τον οποίον ζητείται το παραγοντικό, στη μεταβλητή n του καλούμενου αλγόριθμου Παραγοντικό. Όταν ο τελευταίος τερματίσει, επιστρέφει το αποτέλεσμα nfact.

Οι μεταβλητές n και nfact του αλγόριθμου Συνδυασμοί δεν έχουν καμία σχέση με τις συνώνυμες μεταβλητές του αλγορίθμου Παραγοντικό.

Με τον ίδιο τρόπο που καλείται ο αλγόριθμος Παραγοντικό από τον αλγόριθμο Συνδυασμοί, μπορεί να κληθεί και ο τελευταίος από έναν άλλο αλγόριθμο, όπως π.χ. τον επόμενο.

```
Αλγόριθμος Τράπουλα
Εμφάνισε "Αριθμός Φύλλων:"
Διάβασε fylla
deck ← 52
Κάλεσε Συνδυασμοί(deck, fylla,
                  number)
Εμφάνισε "Οι συνδυασμοί", deck,
         "φύλλων ανά", fylla,
         "είναι:", number
```

Από τη μελέτη του παραδείγματος 2.20 αναδεικνύεται και ο σωστός τρόπος αντιμετώπισης προβλημάτων. Κάθε πρόβλημα διασπάται σε μικρότερα προβλήματα, τα οποία μπορούν να επιλυθούν πιο εύκολα. Ο αλγόριθμος επίλυσης ενός προβλήματος πρέπει να επικεντρώνεται στο πρόβλημα που επιλύει και να αδιαφορεί για το πού θα χρησιμοποιηθεί, καθώς και για την είσοδο και έξοδο των δεδομένων. Με τη χρήση των εντολών Δεδομένα και Αποτελέσματα επιτυγχάνεται πολύ εύκολα η μεταφορά των δεδομένων ενός προβλήματος από/ προς τον αλγόριθμο επίλυσης. Στα επόμενα θα γίνεται αποκλειστική χρήση των εντολών αυτών, εκτός από τις περιπτώσεις που η είσοδος δεδομένων και η έξοδος αποτελεσμάτων είναι το προς επίλυση πρόβλημα.

2.2.7.6 Αναδρομή

Πολλές επιστημονικές εφαρμογές χρησιμοποιούν συναρτήσεις ή σχέσεις γενικότερα που χρησιμοποιούν τις ίδιες στον ορισμό τους. Αυτές οι συναρτήσεις ή σχέσεις ονομάζονται **αναδρομικές** (Recursive).

Παράδειγμα 2.21. Ν Παραγοντικό

Είδαμε στην παράγραφο 2.2.4 τον αναδρομικό ορισμό του Ν!. Με βάση αυτόν τον ορισμό, παρατίθεται ο αναδρομικός αλγόριθμος υπολογισμού του Ν!.

Αλγόριθμος N_Παραγοντικό
Διάβασε N
Κάλεσε Factorial (N, N_Fact)
Εμφάνισε N, "!=", N_Fact
Τέλος N_Παραγοντικό1

Αλγόριθμος Factorial
Δεδομένα // n //
Αν n > 0 **τότε**
  Κάλεσε Factorial(n – 1, Fact)
  Fact ← Fact * n
αλλιώς
  Fact ← 1
Τέλος_αν
Αποτελέσματα // Fact //
Τέλος Factorial

Παράδειγμα 2.22. Μέγιστος Κοινός Διαιρέτης

Παρατίθεται εδώ η αναδρομική εκδοχή του αλγόριθμου του Ευκλείδη

Αλγόριθμος Ευκλείδης
Διάβασε α, β
Κάλεσε ΜΚΔ(α, β)
Τέλος Ευκλείδης

Αλγόριθμος ΜΚΔ
Δεδομένα // x, y //
Αν y = 0 **τότε**
  Εμφάνισε x
αλλιώς
  x ← x **mod** y
  Κάλεσε ΜΚΔ (y, x)
Τέλος_αν
Τέλος ΜΚΔ

Στο παράδειγμα αυτό, ο αλγόριθμος Ευκλείδης καλεί τον αλγόριθμο ΜΚΔ μεταβιβάζοντάς του τις τιμές των μεταβλητών α και β στις μεταβλητές x και y αντίστοιχα. Ο αλγόριθμος ΜΚΔ με την εντολή Κάλεσε καλεί τον εαυτό του δίνοντας άλλες τιμές στις μεταβλητές x και y.

2.2.8 Βασικές αλγοριθμικές λειτουργίες σε δομές δεδομένων

Όπως αναφέρθηκε ήδη, οι δομές δεδομένων διαθέτουν οργανωμένα δεδομένα, στα οποία μπορούν να γίνουν διάφορες επεξεργασίες. Οι βασικές λειτουργίες ή πράξεις επί

των δομών δεδομένων είναι οι ακόλουθες:

- ✓ **Προσπέλαση** (access), πρόσβαση σε δεδομένα με σκοπό την ανάγνωση ή εγγραφή ή μετακίνηση.

- ✓ **Ανάκτηση** (retrieval), η με οποιονδήποτε τρόπο λήψη (ανάγνωση) του περιεχομένου ενός κόμβου.

- ✓ **Αναζήτηση** (searching) ενός συνόλου στοιχείων δεδομένων προκειμένου να εντοπιστούν ένα ή περισσότερα στοιχεία, που έχουν μια δεδομένη ιδιότητα.

- ✓ **Εισαγωγή** (insertion), η προσθήκη ή δημιουργία νέων κόμβων σε μια υπάρχουσα δομή.

- ✓ **Μεταβολή** ή **τροποποίηση** (modification), η αλλαγή του περιεχομένου ενός κόμβου.

- ✓ **Διαγραφή** (deletion) ή **ακύρωση** που συνιστά το αντίθετο της εισαγωγής.

- ✓ **Ταξινόμηση** (sorting), όπου τα στοιχεία μιας δομής διατάσσονται κατά αύξουσα ή φθίνουσα τάξη.

Άλλες λειτουργίες είναι η **συγχώνευση** (merging), κατά την οποία δύο ή περισσότερες ταξινομημένες δομές συνενώνονται σε μια ενιαία δομή, η **προσάρτηση** (append), κατά την οποία μία δομή επικολλάται στο τέλος μιας άλλης, η **αντιγραφή** (copying), κ.ά..

Στην πράξη σπάνια υπάρχουν όλες οι λειτουργίες σε μια δομή. Συνήθως παρατηρείται το φαινόμενο μια δομή δεδομένων να είναι αποδοτικότερη από μια άλλη για κάποια λειτουργία π.χ. την αναζήτηση, αλλά λιγότερο αποδοτική για κάποια άλλη λειτουργία, όπως π.χ. την εισαγωγή. Αυτό εξηγεί τόσο την ύπαρξη διαφορετικών δομών όσο και τη σπουδαιότητα της επιλογής της κατάλληλης δομής κάθε φορά.

Η πιο συνηθισμένη και απλή δομή δεδομένων είναι ο πίνακας. Οι πίνακες υποστηρίζονται από όλες σχεδόν τις γλώσσες προγραμματισμού. Αποτελούνται από ένα σύνολο ομοειδών απλών στοιχείων. Το μέγεθος ενός πίνακα, δηλαδή το πλήθος των στοιχείων που περιέχει, συνήθως είναι σταθερό και προκαθορισμένο. Η αναφορά σε ένα στοιχείο του πίνακα γίνεται με τη χρήση ενός συμβολικού ονόματος που έχει αποδοθεί στον πίνακα, μαζί με ένα ή περισσότερα στοιχεία που ονομάζονται δείκτες (indexes). Για παράδειγμα το πέμπτο στοιχείο του πίνακα Α συμβολίζεται με Α[5].

Στη συνέχεια παρουσιάζονται χαρακτηριστικά παραδείγματα επεξεργασίας πινάκων.

Οι πίνακες με ένα δείκτη λέγονται μονοδιάστατοι, οι πίνακες με 2 δείκτες δισδιάστατοι, ... οι πίνακες με ν δείκτες ν-διάστατοι.

Παράδειγμα 2.23. Εισαγωγή στοιχείων σε πίνακα

➢ Είσοδος δεδομένων αγνώστου πλήθους

Στο επόμενο τμήμα αλγορίθμου εισάγονται θετικοί αριθμοί στο μονοδιάστατο πίνακα Α. Δεν είναι γνωστός ο αριθμός των στοιχείων που θα εισαχθούν, αλλά έχει συμφωνηθεί ότι το τέλος της εισαγωγής θα καθοριστεί από την είσοδο ενός αρνητικού αριθμού (ή γενικότερα μιας

τιμής που δεν μπορεί να ανήκει στο σύνολο τιμών του πίνακα).

```
i ← 0
Διάβασε K
Όσο K ≥ 0 επανάλαβε
        i ← i + 1
        A[i] ← K
        Διάβασε K
Τέλος_επανάληψης
n ← i
```

Εδώ χρησιμοποιείται η εντολή επανάληψης Όσο συνθήκη επανάλαβε. Απαιτεί ένα αρχικό διάβασμα προκειμένου να ενεργοποιηθεί η συνθήκη συνέχισης της επανάληψης. Κάθε θετική τιμή που διαβάζεται καταχωρείται σε επόμενο στοιχείο του πίνακα με τη χρήση του δείκτη i. Στο τέλος φυλάσσεται στη μεταβλητή n η τρέχουσα τιμή του i, ώστε να είναι γνωστό το πλήθος των στοιχείων του πίνακα στη συνέχεια.

- **Είσοδος δεδομένων γνωστού πλήθους**

Όταν είναι γνωστό το πλήθος n των τιμών μπορεί να χρησιμοποιηθεί η εντολή επανάληψης Για … από … μέχρι. Η πιο απλή εκδοχή είναι η επόμενη.

```
Για i από 1 μέχρι n
        Διάβασε A[i]
Τέλος_επανάληψης
```

Σε όλες τις προηγούμενες περιπτώσεις εισαγωγής τιμών σε πίνακα, ο χρήστης του σχετικού προγράμματος μπορεί να υποβοηθείται με την εμφάνιση κατάλληλων μηνυμάτων.

Πριν από κάθε επεξεργασία πίνακα απαιτείται να εισαχθούν δεδομένα σε αυτόν. Η εισαγωγή δεδομένων σε πίνακα μπορεί να είναι επίπονη διαδικασία, ιδιαίτερα αν ο πίνακας είναι μεγάλος. Επί πλέον τα λάθη πληκτρολόγησης είναι πολύ συνηθισμένα και πρέπει να υπάρχει τρόπος διόρθωσης κάποιων τιμών.

Παράδειγμα 2.24. Εκτύπωση πίνακα

Η εκτύπωση των στοιχείων του πί-
νακα, καθώς και της θέσης που
υπάρχει το στοιχείο, επιτυγχάνεται
με τις επόμενες εντολές:

```
Για i από 1 μέχρι n
        Εμφάνισε i, A[i]
Τέλος_επανάληψης
```

Παράδειγμα 2.25. Τροποποίηση στοιχείων πίνακα

Μετά την εισαγωγή στοιχείων και
την εκτύπωση μπορεί να απαιτείται
η διόρθωση ενός ή περισσοτέρων
στοιχείων.

```
Διάβασε i
Όσο i > 0 και i <= n επανάλαβε
        Εμφάνισε A[i]
        Διάβασε A[i]
        Διάβασε i
Τέλος_επανάληψης
```

Στον αλγόριθμο αυτό η επανάληψη εκτελείται όσο δίνεται τιμή του i εντός ορίων. Κάθε φορά εμφανίζεται το στοιχείο A[i] και ζητείται η εισαγωγή μιας άλλης τιμής που την αντικαθιστά.

Μετά την εισαγωγή δεδομένων σε ένα πίνακα μπορεί να ακολουθήσει η επεξεργασία του. Στη συνέχεια παρουσιάζονται μερικές συνηθισμένες επεξεργασίες πινάκων.

Παράδειγμα 2.26. Άθροισμα στοιχείων πίνακα

Δίδεται ο μονοδιάστατος πίνακας Β που περιέχει Ν βαθμούς μαθητών. Να αναπτυχθεί αλγόριθμος, ο οποίος να υπολογίζει και να εμφανίζει το μέσο όρο βαθμολογίας των μαθητών.

Αλγόριθμος Βαθμολογία
Δεδομένα // Β, Ν //
Σ ← 0
Για i **από** 1 **μέχρι** Ν
        Σ ← Σ + Β[i]
Τέλος_επανάληψης
ΜΟ ← Σ / Ν
Εμφάνισε "Μ.Ο.:", ΜΟ
Τέλος Βαθμολογία

Με το βρόχο Για i από 1 μέχρι Ν διατρέχονται όλα τα στοιχεία του πίνακα και κάθε ένα αθροίζεται στη μεταβλητή Σ, η οποία έχει μηδενιστεί πριν από την έναρξη της επανάληψης. Ο Μέσος όρος είναι το πηλίκο του αθροίσματος όλων των στοιχείων δια του πλήθους των στοιχείων.

Για τη δημιουργία προγράμματος που να υλοποιεί τον παραπάνω αλγόριθμο και προκειμένου να γίνει ο έλεγχος ορθότητας, απαιτείται και η εισαγωγή κάποιων δεδομένων. Σύμφωνα με όσα αναφέρθηκαν στις προηγούμενες παραγράφους αυτό μπορεί να γίνει π.χ. με τον επόμενο αλγόριθμο.

```
Αλγόριθμος Επεξεργασία_Πίνακα
Διάβασε n
Για i από 1 μέχρι n
        Διάβασε A[i]
Τέλος_επανάληψης
Κάλεσε Βαθμολογία (A, n)
Τέλος Επεξεργασία_Πίνακα
```

Στις πρώτες γραμμές έχουν γραφεί οι σχετικές εντολές με τις οποίες γίνεται η εισαγωγή δεδομένων στο μονοδιάστατο πίνακα A, ο οποίος έχει n στοιχεία. Στη συνέχεια προκειμένου να βρεθεί ο μέσος όρος των στοιχείων του A, καλείται ο αλγόριθμος Βαθμολογία, που δημιουργήθηκε για το σκοπό αυτό. Κατά την κλήση μεταβιβάζονται στον πίνακα B οι τιμές των στοιχείων του πίνακα A και στη μεταβλητή N η τιμή της μεταβλητής n.

Παράδειγμα 2.27. Μέγιστο στοιχείο πίνακα

Δίδεται ο μονοδιάστατος πίνακας Β που περιέχει Ν βαθμούς μαθητών. Να αναπτυχθεί αλγόριθμος, ο οποίος να βρίσκει και να εμφανίζει την υψηλότερη βαθμολογία.

```
Αλγόριθμος Μέγιστο_πίνακα
Δεδομένα // B, N //
max ← B[1]
Για i από 2 μέχρι N
        Αν B[i] > max τότε
                max ← B[i]
        Τέλος_αν
Τέλος_επανάληψης
Εμφάνισε "Μέγιστο:", max
Τέλος Μέγιστο_πίνακα
```

Στη μεταβλητή max εκχωρείται η τιμή του πρώτου στοιχείου του πίνακα. Στη συνέχεια με το βρόχο Για i από 2 μέχρι Ν εξετάζονται όλα τα υπόλοιπα στοιχεία του πίνακα. Για κάθε ένα, αν είναι μεγαλύτερο από το max, τότε αντικαθιστάται η τιμή του max με το νέο στοιχείο. Δηλαδή στη μεταβλητή max εκχωρείται το εκάστοτε μέγιστο στοιχείο, οπότε στο τέλος της επανάληψης θα έχει το μέγιστο στοιχείο του πίνακα.

Η επανάληψη θα μπορούσε να ξεκινά από 1.

Παράδειγμα 2.28. Επεξεργασία δισδιάστατου πίνακα

Να αναπτυχθεί αλγόριθμος ο οποίος με δεδομένο το πλήθος των μαθητών της Γ' γυμνασίου ενός σχολείου και τους βαθμούς των μαθητών στη γυμναστική, σε καθένα από τα 3 τρίμηνα, να υπολογίζει και να επιστρέφει το μέσο όρο βαθμολογίας κάθε μαθητή και το συνολικό μέσο όρο.

Για την εισαγωγή δεδομένων σε ένα δισδιάστατο πίνακα Α που έχει Ν γραμμές και Μ στήλες, θα μπορούσε να αναπτυχθεί το ακόλουθο τμήμα αλγορίθμου:

```
Για i από 1 μέχρι N
  Για j από 1 μέχρι M
    Διάβασε A[i, j]
  Τέλος_επανάληψης
Τέλος_επανάληψης
```

Ο αλγόριθμος αθροίζει τα στοιχεία κάθε γραμμής ενός δισδιάστατου πίνακα N γραμμών και 3 στηλών.

Ο πίνακας B έχει N γραμμές και 3 στήλες. Σε κάθε μία γραμμή υπάρχουν οι βαθμολογίες των τριών τριμήνων για κάθε ένα μαθητή.

```
Αλγόριθμος Αθροίσματα
Δεδομένα // B, N //
Σ ← 0
Για i από 1 μέχρι N
   ΣΜ ← 0
```

```
   Για j από 1 μέχρι 3
      ΣΜ ← ΣΜ + Β[i, j]
   Τέλος_επανάληψης
   ΜΟΒ[i]← ΣΜ / 3
   Σ ← Σ + ΣΜ
Τέλος_επανάληψης
ΜΟ ← Σ / (Ν * 3)
Αποτελέσματα // ΜΟΒ, ΜΟ //
Τέλος Αθροίσματα
```

Με το διπλό βρόχο Για i και Για j σαρώνονται κατά γραμμή όλα τα στοιχεία του πίνακα Β. Σε κάθε μία γραμμή, δηλαδή για κάθε μία τιμή του i, μηδενίζεται το ΣΜ (άθροισμα βαθμών μαθητή) και ακολούθως με τον εσωτερικό βρόχο αθροίζονται όλα τα στοιχεία της γραμμής αυτής. Όταν ολοκληρωθεί ο εσωτερικός βρόχος,

στο MOB[i] βρίσκεται ο μέσος όρος των στοιχείων της εκάστοτε γραμμής, δηλαδή ο μέσος όρος κάθε μαθητή στα τρία τρίμηνα. Το άθροισμα που υπολογίστηκε ανά γραμμή αξιοποιείται και αυξάνει τη μεταβλητή Σ κατά το άθροισμα γραμμής, η οποία στο τέλος θα έχει το άθροισμα όλων των στοιχείων του πίνακα.

Ο μέσος όρος είναι το πηλίκο του αθροίσματος δια του πλήθους των στοιχείων του δισδιάστατου πίνακα. Το πλήθος των στοιχείων του είναι N*3.

Παράδειγμα 2.29. Σειριακή αναζήτηση

Σε ένα μονοδιάστατο πίνακα είναι καταχωρημένα τα ονόματα των σχολείων που συμμετέχουν σε έναν διαγωνισμό επιχειρηματικότητας και έχουν κατασκευάσει ένα χώρο παρουσίασης του επιχειρηματικού τους σχεδίου (που εν συντομία θα λέγεται περίπτερο). Ο αριθμός του περιπτέρου κάθε σχολείου είναι η θέση του σχολείου στον πίνακα. Δηλαδή στο πρώτο κελί του πίνακα είναι το όνομα του σχολείου που έχει το περίπτερο με αριθμό 1, στο δεύτερο κελί του πίνακα είναι το όνομα του σχολείου που έχει το περίπτερο με αριθμό 2 κ.ο.κ. Να αναπτυχθεί αλγόριθμος ο οποίος με δεδομένο το πλήθος των σχολείων

που συμμετέχουν στο διαγωνισμό με περίπτερο και τον πίνακα με τα ονόματα των σχολείων, να διαβάζει το όνομα ενός σχολείου και να ψάχνει στον πίνακα, αν υπάρχει αυτό το σχολείο. Ο αλγόριθμος θα επιστρέφει το αποτέλεσμα της αναζήτησης, δηλαδή αν υπάρχει ή όχι το αναζητούμενο σχολείο, και αν υπάρχει τη θέση του στον πίνακα, αλλιώς ως θέση την τιμή μηδέν.

Κάθε σχολείο μπορεί να έχει ένα περίπτερο.

Αλγόριθμος Αναζήτηση
Δεδομένα // Α, Ν //
Εμφάνισε “Αναζητούμενο σχολείο:”
Διάβασε Κ

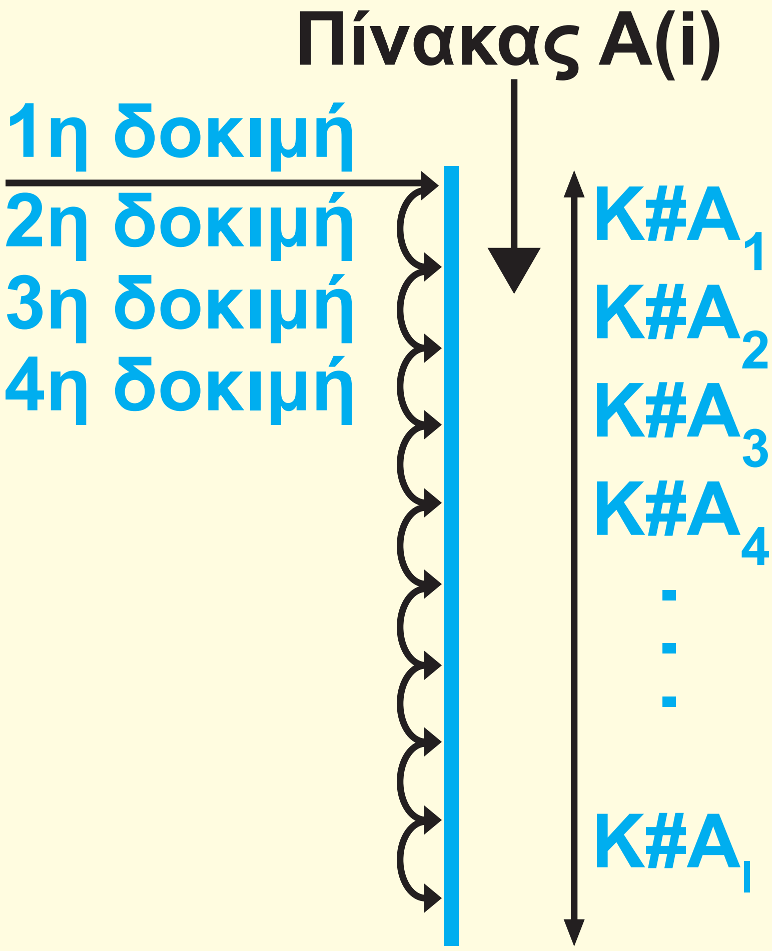
```
i ← 1
Βρέθηκε ← Ψευδής
Θέση ← 0
Όσο i ≤ N και Βρέθηκε = Ψευδής
επανάλαβε
        Αν K = A[i] τότε
                Θέση ← i
                Βρέθηκε ← Αληθής
        αλλιώς
                i ← i + 1
        Τέλος_αν
Τέλος_επανάληψης
Αποτελέσματα // Βρέθηκε, Θέση //
Τέλος Αναζήτηση
```

Τα αποτελέσματα είναι η μεταβλητή Βρέθηκε, η οποία, αν είναι αληθής σημαίνει ότι η αναζήτηση ήταν επιτυχής και η μεταβλητή Θέση, που έχει τη θέση του στοιχείου στον πίνακα, εφ’ όσον βρέθηκε, αλλιώς θα έχουν την τιμή Ψευδής και μηδέν αντίστοιχα. Η τιμή μηδέν δεν μπορεί να είναι θέση πίνακα.

Στον αλγόριθμο είναι δεδομένος ο πίνακας Α και το πλήθος των στοιχείων του (Ν).

Η λογική μεταβλητή Βρέθηκε έχει την αρχική τιμή Ψευδής, όπου θεωρείται ότι δεν υπάρχει το ζητούμενο σχολείο και η μεταβλητή Θέση έχει την αρχική τιμή 0 για τον ίδιο λόγο.

Η επανάληψη του αλγορίθμου εκτελείται όσο δεν τελείωσε η σάρωση του πίνακα (i ≤ N) και όσο δεν βρέθηκε το στοιχείο (Βρέθηκε = Ψευδής).



Εικόνα 2.24. Σειριακή αναζήτηση.

Παράδειγμα 2.30. Ταξινόμηση με επιλογή

Να αναπτυχθεί αλγόριθμος ο οποίος με δεδομένο το πλήθος των μαθητών της Γ’ γυμνασίου ενός σχολείου και τον τελικό βαθμό του κάθε μαθητή στο μάθημα «Πληροφορική», να ταξινομεί τον πίνακα των βαθμών σε αύξουσα τάξη, από το μικρότερο στο μεγαλύτερο βαθμό. Στη συνέχεια να επιστρέφει τον ταξινομημένο πίνακα.

```
Αλγόριθμος Ταξινόμηση_με_επιλογή
Δεδομένα // A, N //
Για i από 1 μέχρι N
        j ← i
        min ← A[j]
        Για k από i + 1 μέχρι N
                Αν A[k] < min τότε
                        j ← k
                        min ← A[j]
                Τέλος_αν
        Τέλος_επανάληψης
        temp ← A[i]
        A[i] ← A[j]
        A[j] ← temp
```

```
Τέλος_επανάληψης
Αποτελέσματα // Α //
Τέλος Ταξινόμηση_με_επιλογή
```

Η μέθοδος της ταξινόμησης με επιλογή βασίζεται στα ακόλουθα τρία βήματα:

α. επιλογή του στοιχείου με την ελάχιστη τιμή.

β. ανταλλαγή του με το πρώτο στοιχείο.

γ. επανάληψη των (α), (β) με τα υπόλοιπα στοιχεία.

Ο αλγόριθμος ταξινόμησης με επιλογή είναι πολυπλοκότητας $O(n^2)$

2.2.9 Εκσφαλμάτωση σε λογικά λάθη

Ένας αλγόριθμος χρειάζεται να δοκιμαστεί σε διαφορετικές συνθήκες και με ποικιλία δεδομένων ώστε να συγκριθούν τα παραγόμενα αποτελέσματα με τα αναμενόμενα, και να προβλεφτούν απρόσμενες καταστάσεις λάθους.

Ως **εκσφαλμάτωση** των λογικών λαθών ενός αλγορίθμου προσδιορίζεται η διαδικασία εύρεσης των λογικών λαθών που υπάρχουν σε αυτόν.

Το πρώτο βήμα στην εύρεση των λογικών λαθών είναι να γίνει προσπάθεια εντοπισμού των πιθανών πηγών τους. Για να επιτευχθεί αυτό χρειάζεται να δώσετε προσοχή στα αποτελέσματα που έχει ως έξοδο ο αλγόριθμος και να σκεφτείτε πιθανές αιτίες που μπορεί να οδήγησαν σε τέτοιου είδους αποτελέσματα.

Η ανίχνευση τέτοιων λαθών δεν είναι δυνατό να πραγματοποιηθεί από κάποιο εργαλείο του υπολογιστή και διαπιστώνονται μόνο με τη διαδικασία ελέγχου και την ανάλυση των αποτελεσμάτων του. Ένα λογικό λάθος είναι ένα λάθος που, ενώ εκτελείται ο αλγόριθμος, τα αποτελέσματά του δεν είναι σωστά. Αυτό μπορεί να οφείλεται είτε σε λανθασμένη προσέγγιση για το πώς θα λυθεί το πρόβλημα, είτε σε λανθασμένη υλοποίηση της προσέγγισης που επιλέχθηκε. Κατά γενική ομολογία, τα λογικά λάθη είναι δύσκολο να εντοπιστούν.

Η εκτέλεση του αλγορίθμου με το χέρι είναι πολύ χρήσιμη διαδικασία για τον εντοπισμό των λογικών λαθών.

Για το λόγο αυτό χρειάζεται να δοκιμάζετε εξονυχιστικά τον αλγόριθμό σας με διάφορα παραδείγματα, απλά, σύνθετα και ασυνήθιστα, για να βεβαιωθείτε ότι είναι σωστός.

Παράδειγμα 2.31. Λανθασμένος αλγόριθμος

Με σκοπό να αναπτυχθεί αλγόριθμος ο οποίος με δεδομένες τις τιμές δύο μεταβλητών, θα αντιμεταθέτει το περιεχόμενο των δύο μεταβλητών και θα επιστρέφει ως αποτέλεσμα το περιεχόμενο των μεταβλητών μετά την αντιμετάθεση γράφτηκε ο ακόλουθος λανθασμένος αλγόριθμος σε ψευδογλώσσα. Ζητείται να εντοπιστούν τα λογικά λάθη.

1. **Αλγόριθμος** Αντιμετάθεση
2. **Δεδομένα** // α, β //
3. α ← β
4. β ← α
5. **Αποτελέσματα** // α, β //
6. **Τέλος** Αντιμετάθεση

Απάντηση

Το πρώτο βήμα είναι η εκτέλεση του αλγορίθμου με πίνακα παρακολούθησης τιμών για να ελεγχθεί η λειτουργία του.

Αριθμός Εντολής	α	β	Έξοδος
2	8	12	
3	12		
4		12	
5			12 12

Από την εκτέλεση φαίνεται ότι εκτυπώνει σωστά την τιμή της μεταβλητής α, αλλά όχι της β. Αφού διαπιστωθεί το λάθος χρειάζεται να προσδιοριστεί η απαιτούμενη διόρθωση. Από τον αλγόριθμο απουσιάζει η μεταβλητή στην οποία θα εκχωρούταν προσωρινά η τιμή μίας εκ των δύο μεταβλητών.

Αν κάποιο από τα παραδείγματα δεν δίνει το αναμενόμενο αποτέλεσμα τότε υπάρχει παράλειψη στον κώδικα και δεν έχει καλυφθεί η σχετική περίπτωση.

Μόλις βρείτε τι φταίει, εμπλουτίζετε την αρχική σας σχεδίαση ώστε να καλύψετε και αυτό το είδος περιπτώσεων και κάνετε τις απαραίτητες προσθήκες.

Η διαδικασία εκσφαλμάτωσης περιλαμβάνει:

- Τη διαπίστωση του είδους του λάθους.

- Την ανεύρεση του ανεξάρτητου τμήματος του αλγορίθμου που εκτελεί τη λανθασμένη λειτουργία.

- Την ανεύρεση του λάθους μέσα σε αυτό το ανεξάρτητο τμήμα του αλγορίθμου.

2.2.10 Τεκμηρίωση

Η τεκμηρίωση δεν αποτελεί μια ξεχωριστή φάση στην ανάπτυξη ενός αλγορίθμου, αλλά μία παράλληλη διαδικασία που συμπληρώνει όλα τα στάδια ανάπτυξής του.

Με τον όρο **τεκμηρίωση** εννοεί-ται το σύνολο του γραπτού υλι-κού που περιγράφει τα συστατικά μέρη και τις λειτουργίες του αλγο-ρίθμου ή τις τροποποιήσεις που έγιναν σε έναν αλγόριθμο.

Οι λόγοι τεκμηρίωσης είναι:

- η ευκολία ανάπτυξης αλγορίθμου σε περίπτωση που ζητείται αλγό-ριθμος με παρόμοιες λειτουργίες με υπάρχοντα αλγόριθμο.
- Η ευκολία στις τροποποιήσεις ενός υπάρχοντος αλγορίθμου.
- Ο περιορισμός των ελέγχων μόνο σε συγκεκριμένα σημεία, αφού

για τα υπόλοιπα θα υπάρχει αρχείο δοκιμής μαζί με αποτελέσματα και συνοδευτικά σχόλια.

Η τεκμηρίωση μπορεί να διακριθεί:

- Τεκμηρίωση ανάπτυξης. Αναφέρεται στις προδιαγραφές του προβλήματος, τι πρόκειται να κάνει ο αλγόριθμος και τη σύνδεσή του με άλλους αλγορίθμους.

- Τεκμηρίωση ελέγχου. Αναφέρεται στα δεδομένα ελέγχου που χρησιμοποιήθηκαν προκειμένου να δοκιμαστεί ο αλγόριθμος, αν έχουν διερευνηθεί οι ακραίες περιπτώσεις και ποιες είναι αυτές και τέλος αν έχουν καθιερωθεί κάποιες και ποιες συμβάσεις για τη λύση του προβλήματος.

- **Τεκμηρίωση αλγορίθμου.** Αναφέρεται στον τρόπο με τον οποίο έχουν λυθεί τα επιμέρους προβλήματα, τις υποθέσεις που έχουν γίνει και τους περιορισμούς που έχουν τεθεί. Ακόμα πληροφορίες για «ειδικές» τεχνικές που έχουν χρησιμοποιηθεί. Πρέπει σε κάθε βήμα να αναφερθεί με λεπτομέρεια, πολλές φορές κουραστική, αλλά δυστυχώς αναγκαία, η λύση που χρησιμοποιήθηκε και τα δεδομένα με τα οποία ελέγχθηκε. Ένα υπόμνημα των μεταβλητών που χρησιμοποιούνται είναι χρήσιμο για την τεκμηρίωση αλλά και για την αρχική ανάπτυξη του αλγόριθμου.

Η απλούστερη και πλέον στοιχειώδης τεκμηρίωση αυτής της μορφής γίνεται με την εισαγωγή γραμμών σχολίων μέσα στον αλγόριθμο.

Ανακεφαλαίωση

Στις παραγράφους του κεφαλαίου παρουσιάστηκαν η έννοια του αλγορίθμου μαζί με τα χαρακτηριστικά που χρειάζεται να έχει, οι τρόποι παράστασης αλγορίθμων, τα συστατικά τους μέρη, οι δομές δεδομένων και διατυπώθηκαν βασικά στοιχεία ανάλυσης και θεωρίας αλγορίθμων. Επιπλέον, παρουσιάστηκε η ψευδογλώσσα ως βασικότερος τρόπος ανάπτυξης αλγορίθμων, προκειμένου να υπάρχει ανεξαρτησία από τις γλώσσες προγραμματισμού. Παρουσιάστηκαν πολλά παραδείγματα απλών αλγορίθμων, ενώ αναπτύχθηκαν αλγόριθμοι αναζήτησης και ταξινόμησης. Τέλος, στο κεφάλαιο αυτό, επιχειρήθηκε να αναδειχθεί ότι οι αλγόριθμοι και οι δομές δεδομένων έχουν ισχυρότατη σχέση και αποτελούν τα βασικά συστατικά για την ανάπτυξη προγραμμάτων.

Λέξεις κλειδιά
Αλγόριθμος, Χαρακτηριστικά αλγορίθμου, Ψευδογλώσσα, Διάγραμμα ροής, Μεταβλητές, Εκφράσεις, Εντολή εκχώρησης, Δομή ακολουθίας, Δομή επιλογής, Δομή επανάληψης,

Δομές δεδομένων, Πίνακας, Ουρά, Στοίβα, Γράφος, Δένδρο, Αναζήτηση, Ταξινόμηση, Εκσφαλμάτωση, Τεκμηρίωση.

Ερωτήσεις - Θέματα προς συζήτηση - Δραστηριότητες

1. Τι είναι αλγόριθμος; Καταγράψτε ή συζητήστε με τους συμμαθητές σας έναν αλγόριθμο.

2. Μέσα από παιχνίδι ρόλων να υλοποιήσετε και να εκτελέσετε τον αλγόριθμο του παραδείγματος 2.5 σειριακά και παράλληλα.

3. Με ποιους τρόπους γίνεται η αναπαράσταση αλγορίθμων;

4. Ποια η διαφορά δεδομένου και πληροφορίας; Να δώσετε ένα παράδειγμα.

5. Ποιοι είναι οι συνηθέστεροι τύποι δεδομένων; Ποιος τύπος λαμβάνει τις λιγότερες τιμές;

6. Τι είναι η δομή δεδομένων;

7. Ποιες δομές δεδομένων είναι μη γραμμικές;

8. Ποια είναι τα είδη των τελεστών;

9. Ποια είναι η χρήση της δομής ακολουθίας; Να δώσετε έναν αλγόριθμο σε ψευδογλώσσα με τη χρήση της δομής ακολουθίας.

10. Να περιγράψετε τη λειτουργία των τριών εντολών επιλογής.

11. Να περιγράψετε τη λειτουργία των τριών εντολών επανάληψης.

12. Να εξηγήσετε πότε δεν μπορεί να χρησιμοποιηθεί η εντολή επανάληψης Για.

13. Ποιες είναι οι λειτουργίες ή πράξεις επί των δομών δεδομένων;

14. Να περιγράψετε τον αλγόριθμο ταξινόμησης επιλογής σε πίνακα.

15. Σημειώστε τις σωστές απαντήσεις

Α. **Ποια από τα παρακάτω αποτελούν αριθμητική σταθερά;**

- **i.** **2009**
- **ii.** **"2009"**
- **iii.** **Εμφάνισε**
- **iv.** **"Αλγόριθμος"**

Β. **Η λογική πράξη ή μεταξύ δύο προτάσεων είναι αληθής όταν:**

- **i.** **Οποιαδήποτε από τις προτάσεις είναι αληθής**
- **ii.** **Η πρώτη πρόταση είναι αληθής**
- **iii.** **Η πρώτη πρόταση είναι ψευδής**
- **iv.** **Και οι δύο προτάσεις είναι αληθείς**

Γ. Αν Α και Β ακέραιες μεταβλητές, ποιες από τις παρακάτω εκφράσεις είναι αριθμητικές;

i. $A + B \wedge 2$

ii. $A > B + 3$

iii. $A \text{ div } 3 \wedge B$

iv. A και $B > 3$

16. Να μετατρέψετε σε εντολές εκχώρησης τις παρακάτω φράσεις:

Α. Η μεταβλητή α έχει διπλάσια τιμή από τη μεταβλητή β

Β. Η μεταβλητή ΜΟ είναι ο μέσος όρος των α, β, γ

Γ. Η μεταβλητή β αυξάνεται κατά 2

Δ. Η μεταβλητή i μειώνεται κατά α και β

Ε. Η μεταβλητή i είναι το μισό του αθροίσματος των α και β

17. Να επιλέξετε την τιμή της y που είναι αποτέλεσμα κάθε εντολής.

Εντολές εκχώρησης	Τιμή της y
i. y←(A_M(7 / 2) mod 3) + 1 ii. y←10 div 9 – 10 mod 9 iii. y ←10 mod 2 + A_T(2 – 3) iv. y ←A_M(9 / 2 / 3)	A. 1 B. 0

18. Αντιστοιχίστε τις εκφράσεις της στήλης Α με τις λογικές σταθερές της στήλης Β με δεδομένο ότι $\alpha = 10$, $\beta = 5$ και $\gamma = 3$.

Στήλη Α	Στήλη Β
i. $\alpha \neq \beta$ **και** $(\gamma - \beta) < 0$	Α. Αληθής
ii. $(\alpha > \beta$ **ή** $\alpha > \gamma)$ **και** $(\gamma > \beta)$	Β. Ψευδής
iii. $\alpha > \beta$ **ή** $\alpha > \gamma$ **και** $\gamma > \beta$	
iv. **όχι** $(\alpha > \beta)$ **ή** $\gamma \leq \beta$	
v. $\alpha > \beta$ **ή** $(\beta - \gamma) > 0$ **και** $\alpha < 0$	
vi. $\alpha > \beta$ **ή** $(\beta + 3) < \gamma$ **και** $\alpha < \gamma$	

19. Τι εμφανίζουν τα επόμενα τμήματα αλγορίθμων;

```
S ← 0
i ← 1
Όσο i ≤ 9 επανάλαβε
    i ← i + 2
    S ← S + i
Τέλος_επανάληψης
Εμφάνισε S
```

```
S ← 0
i ← 1
Όσο i ≤ 9 επανάλαβε
    S ← S + i
    i ← i + 2
Τέλος_επανάληψης
Εμφάνισε S
```

20. Ο νόμος του Νεύτωνα για τη βαρύτητα λέει ότι κάθε σώμα στο σύμπαν έλκει κάθε άλλο σώμα με δύναμη που δίνεται από τον τύπο $F = G \cdot \frac{m1 \cdot m2}{r^2}$, όπου m1 και m2 είναι οι μάζες των δύο σωμάτων (σε κιλά), r η απόσταση μεταξύ τους (σε μέτρα) και G είναι η παγκόσμια βαρυτική σταθερά. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει τις δύο μάζες, την απόσταση μεταξύ τους και θα υπολογίζει και θα εκτυπώνει τη δύναμη. Δίνεται ότι $G = 6{,}67 \times 10^{-11}\ N\ m^2\ kg^{-2}$.

21. Στην περίοδο των εκπτώσεων αγοράσατε ένα ποδήλατο με έκπτωση 25%. Το ποσό που δώσατε για το ποδήλατο ήταν 100 ευρώ. Να αναπτύξετε αλγόριθμο ο οποίος θα υπολογίζει την αρχική τιμή του ποδηλάτου.

22. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει έναν αριθμό και θα υπολογίζει και θα εμφανίζει το γινόμενο αυτού του αριθμού επί το τελευταίο ψηφίο του. Θεωρήστε ότι ο αριθμός είναι θετικός και ακέραιος.

23. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει έναν πραγματικό αριθμό με 2 δεκαδικά ψηφία και θα τον στρογγυλοποιεί στον πλησιέστερο ακέραιο. Για παράδειγμα, αν διαβαστεί ο αριθμός 4,23, να

εμφανίζει 4, ενώ αν είναι ο 4,70 να εμφανίζει 5.

24. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει έναν ακέραιο αριθμό και θα υπολογίζει και θα εμφανίζει τον επόμενο άρτιο.

25. Σε έναν λογαριασμό τραπέζης παρέχεται κλιμακωτά το ακόλουθο επιτόκιο:

Ποσό	Επιτόκιο
≤ 5.000	1,8% το έτος
> 5.000	1,5% το έτος

Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει το ποσό χρημάτων που έχει ο λογαριασμός και θα υπολογίζει και θα εμφανίζει τον τόκο που θα λάβει μετά από ένα έτος, καθώς και το συνολικό ποσό χρημάτων μαζί με τον τόκο.

26. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει το τρέχον έτος και αν αυτό είναι από το 2001 μέχρι και το 2099 να εμφανίζει το μήνυμα «21ος αιώνας». Αν το έτος είναι από το 2002 και πάνω, να εμφανίζει το μήνυμα «Χρήση του €».

27. Ένα επιστημονικό σωματείο έχει 1.200 μέλη. Η γενική συνέλευση του σωματείου είναι σε

απαρτία όταν είναι παρόν το 1/3 των μελών του. Για να υπερψηφιστεί μια πρόταση, θα πρέπει περισσότεροι από το 1/2 των παρόντων μελών να ψηφίσουν υπέρ. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει τον αριθμό των παρόντων μελών και αν ο αριθμός επιτρέπει την πραγματοποίηση της ψηφοφορίας, θα διαβάζει τον αριθμό αυτών που ψήφισαν υπέρ της πρότασης και θα εμφανίζει το αποτέλεσμα της ψηφοφορίας, δηλαδή αν υπερψηφίστηκε, αν καταψηφίστηκε ή αν δεν μπορεί να ψηφιστεί.

28. Ένας συνδρομητής μιας εταιρείας κινητής τηλεφωνίας έχει επιλέξει ένα πρόγραμμα με πάγιο

50 ευρώ τον μήνα. Στο πρόγραμμα δικαιούται τις ακόλουθες παροχές:

Παροχές	Πλήθος
Λεπτά ομιλίας / μήνα	1.000
SMS/μήνα	1.000
MB/μήνα	1.000

Ωστόσο, αν ξεπεράσει τον αριθμό 1.000 σε κάποια από τις παραπάνω παροχές, τότε χρεώνεται ως εξής για κάθε παροχή που ξεπερνάει τα 1.000:

Επιπλέον	Πλήθος
Κλήσεις ομιλίας	0,0055 € / δευτερόλεπτο
SMS	0,08 €/SMS
MB	0,05 €/MB

Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει τα λεπτά ομιλίας, το πλήθος των SMS, το πλήθος των MB και ανάλογα θα εμφανίζει τη μηνιαία χρέωση του καταναλωτή.

29. Να αναπτύξετε αλγόριθμο ο οποίος θα εκτυπώνει τις τιμές της συνάρτησης $f(x) = 4\log(5 + e^{3x+2})$ όταν το x παίρνει τις τιμές στο διάστημα [10, 50] με βήμα 0,5.

30. Να αναπτύξετε αλγόριθμο ο οποίος θα εμφανίζει όλους τους τριψήφιους αριθμούς που το άθροισμα των ψηφίων τους είναι μεγαλύτερο ή ίσο του 12.

31. Να αναπτύξετε αλγόριθμο ο οποίος θα υπολογίζει το ημίτονο ενός αριθμού x με βάση τον επόμενο τύπο

$$ημ(x) = \sum_{n=0}^{49} (-1)^n \frac{x^{2n+1}}{(2n+1)!} \quad ή$$

$$ημ(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \ldots - \frac{x^{49}}{49!}$$

32. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει γράμματα μέχρι να βρει τρεις φορές το γράμμα Α. Όταν σταματήσει το διάβασμα γραμμάτων, ο αλγόριθμος θα εκτυπώνει πόσα συνολικά γράμματα διαβάστηκαν.

33. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει αριθμούς, μέχρι να διαβαστεί ο αριθμός μηδέν. Ο αλγόριθμος θα εκτυπώνει το άθροισμα και το πλήθος των αριθμών που δόθηκαν και ήταν μεγαλύτεροι του 50.

34. Ένα ψηφιακό φωτογραφικό άλμπουμ έχει αποθηκευτικό χώρο N Mbytes. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει το μέγεθος της κάθε φωτογραφίας που επιχειρείται να αποθηκευτεί στο άλμπουμ, μέχρι το άλμπουμ να μη χωράει άλλη φωτογραφία. Ο αλγόριθμος θα

επαναλαμβάνεται και θα σταματά αν το μέγεθος της φωτογραφίας που προσπαθεί κάποιος να αποθηκεύσει είναι μεγαλύτερο από τον διαθέσιμο χώρο του άλμπουμ. Όταν η εισαγωγή φωτογραφιών σταματήσει, ο αλγόριθμος θα εκτυπώνει το μήνυμα «Δεν χωράει». Στην περίπτωση που περίσσεψε χώρος να τον εκτυπώνει. Τέλος, να εκτυπώνει το πλήθος των φωτογραφιών που αποθηκεύτηκαν.

35. Τροποποιείστε τον αλγόριθμο του παραδείγματος 2.26, ώστε να βρίσκει το γινόμενο των στοιχείων του πίνακα.

36. Να αναπτύξετε αλγόριθμο ο οποίος:

Α. Θα διαβάζει το πλήθος των μαθητών ενός λυκείου.

Β. Θα διαβάζει το μέσο όρο βαθμολογίας κάθε μαθητή ο οποίος θα εισάγεται σε πίνακα. Ο μέσος όρος βαθμολογίας κάθε μαθητή θα πρέπει να ελέγχεται ώστε να είναι μεγαλύτερος ή ίσος του (ένα) 1 και μικρότερος ή ίσος του είκοσι (20).

Γ. Θα εμφανίζει σε ποια θέση βρίσκεται και ποιος είναι ο μικρότερος μέσος όρος. Θεωρήστε ότι είναι μοναδικός.

Δ. Θα εμφανίζει το μέσο όρο όλων των μαθητών.

Ε. Θα εμφανίζει με κατάλληλα μηνύματα το ποσοστό των μαθητών με μέσο όρο μεγαλύτερο του 18.

37. Ενοποιείστε τους αλγορίθμους των παραδειγμάτων 2.23, 2.24 και 2.25 σε έναν ενιαίο αλγόριθμο, στον οποίο να προτάξετε ένα μενού επιλογής μιας λειτουργίας. (βλ. παρ. 2.17).

38. Στο πλαίσιο των εικονικών μαθητικών επιχειρήσεων, να αναπτύξετε αλγόριθμο ο οποίος:

Α. Θα διαβάζει και θα αποθηκεύει σε πίνακα τις εισπράξεις μιας επιχείρησης για κάθε μήνα ενός σχολικού έτους.

Β. θα ταξινομεί τον πίνακα των εισπράξεων σε φθίνουσα σειρά και θα εμφανίζει τον ταξινομημένο πίνακα.

Γ. θα εμφανίζει τη μικρότερη

και τη μεγαλύτερη είσπραξη της επιχείρησης.

39. Τροποποιείστε τον αλγόριθμο του παραδείγματος 2.28, ώστε να βρίσκει το μέσο όρο κάθε μαθήματος (υποδ. Απαιτείται ο υπολογισμός αθροισμάτων κατά στήλη).

40. Ο επόμενος αλγόριθμος εκτελεί σειριακή αναζήτηση του στοιχείου Κ στον πίνακα Α. Εντοπίστε τις διαφορές με τον αλγόριθμο του παραδείγματος 2.29. Ποιος είναι προτιμότερος και γιατί; Να λάβετε υπόψη την περίπτωση επιτυχημένης και αποτυχημένης αναζήτησης.

```
Αλγόριθμος Αναζήτηση2
Δεδομένα // A, N, K //
Βρέθηκε ← Ψευδής
Θέση ← 0
Για  i από 1 μέχρι N
    Αν K = A[i] τότε
            Θέση ← i
            Βρέθηκε ← Αληθής
    Τέλος_αν
Τέλος_επανάληψης
Αποτελέσματα // Βρέθηκε, Θέση //
Τέλος Αναζήτηση2
```

41. Στον αλγόριθμο του παραδείγματος 2.29 είναι γνωστό από την εκφώνηση, ότι δεν μπορεί να υπάρχει στον πίνακα δεύτερη φορά το ίδιο σχολείο. Αν σε έναν πίνακα δεν είναι γνωστό αν υπάρχουν ή όχι επαναλήψεις των ίδιων στοιχείων, πώς πρέπει να αντιμετωπιστεί το πρόβλημα της αναζήτησης; Δώστε ένα σχετικό αλγόριθμο.

42. Αν ο πίνακας Α του παραδείγματος 2.29 ήταν ταξινομημένος, μήπως αυτή η επί πλέον γνώση μπορεί να επιφέρει κάποια βελτίωση στον αλγόριθμο σειριακής αναζήτησης; Αν ναι, εκπονήστε τον τροποποιημένο αλγόριθμο.

43. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει τις δικαιολογημένες και τις αδικαιολόγητες απουσίες ενός μαθητή καθώς και τον μέσο όρο του στα προφορικά και θα καλεί αλγόριθμο

ο οποίος θα δέχεται τα παραπάνω στοιχεία και θα επιστρέφει το μήνυμα:

- «Έχει δικαίωμα εξέτασης τον Ιούνιο», αν ο μαθητής έχει μέχρι 64 απουσίες ή έχει μέχρι 114 απουσίες από τις οποίες οι αδικαιολόγητες δεν ξεπερνούν τις 64 ή έχει μέχρι 164 απουσίες από τις οποίες οι αδικαιολόγητες δεν ξεπερνούν τις 64 και ο μέσος όρος του στα προφορικά είναι πάνω από 15.

- «Έχει δικαίωμα εξέτασης τον Σεπτέμβριο», αν ο μαθητής έχει πάνω από 64 και μέχρι 114 απουσίες και οι αδικαιολόγητες ξεπερνούν τις 64 ή έχει πάνω από 114 και μέχρι 164 απουσίες από τις οποίες οι αδικαιολόγητες δεν ξεπερνούν τις 64 αλλά ο μέσος όρος του στα προφορικά δεν είναι πάνω από 15.

- «Επανάληψη χρονιάς», σε κάθε άλλη περίπτωση.

Ο καλών αλγόριθμος θα εκτυπώνει το μήνυμα.

44. Να αναπτύξετε αλγόριθμο ο οποίος θα διαβάζει έναν ακέραιο θετικό αριθμό, και θα καλεί αλγόριθμο ο οποίος θα υπολογίζει το πλήθος των ψηφίων του αριθμού. Ο καλούμενος αλγόριθμος,

θα επιστρέφει το πλήθος των ψηφίων του αριθμού. Ο καλών αλγόριθμος θα εκτυπώνει το πλήθος.

45. Να αναπτύξετε αλγόριθμο ο οποίος:
   Α. Θα διαβάζει και θα καταχωρίζει σε έναν πίνακα 1000 στοιχείων ονόματα.
   Β. Θα διαβάζει ένα ζητούμενο όνομα.
   Γ. Θα καλεί αλγόριθμο, ο οποίος θα αναζητά στον πίνακα το όνομα που διαβάστηκε στο προηγούμενο ερώτημα και θα επιστρέφει το πλήθος των ατόμων που υπάρχουν με το όνομα αυτό. Ο καλών αλγόριθμος θα εκτυπώνει το πλήθος.

46. Να εκτελέσετε το παρακάτω τμήμα αλγορίθμου

```
x ← 20
Σ ← 0
Για α από 1 μέχρι 10 με_βήμα 3
        x ← x – α
        Σ ← Σ + x
        Αν Σ mod 2 = 0 τότε
                x ← x + 1
        Τέλος_αν
Τέλος_επανάληψης
Αποτελέσματα // x //
```

47. Το ακόλουθο τμήμα αλγορίθμου αναπτύχθηκε για να υπολογίζει το άθροισμα 5 αριθμών. Ωστόσο περιέχει λογικά λάθη. Να εντοπίσετε τα λογικά λάθη και να τον διορθώσετε.

```
Διάβασε x
Σ ← 0
Για i από 0 μέχρι 5
        Διάβασε x
        Σ ← Σ + x
Τέλος_επανάληψης
Εμφάνισε Σ + x
```

48. Το ακόλουθο τμήμα αλγορίθμου αναπτύχθηκε για να ελέγχει αν ένας αριθμός είναι θετικός. Ωστόσο περιέχει λογικό λάθος. Να το εντοπίσετε.

```
Επανάλαβε
        Διάβασε α
Μέχρις_ότου α < 0
```

Περιεχόμενα

Βάσει του ν. 3966/2011 τα διδακτικά βιβλία του Δημοτικού, του Γυμνασίου, του Λυκείου, των ΕΠΑ.Λ. και των ΕΠΑ.Σ. τυπώνονται από το ΙΤΥΕ - ΔΙΟΦΑΝΤΟΣ και διανέμονται δωρεάν στα Δημόσια Σχολεία. Τα βιβλία μπορεί να διατίθενται προς πώληση, όταν φέρουν στη δεξιά κάτω γωνία του εμπροσθόφυλλου ένδειξη «ΔΙΑΤΙΘΕΤΑΙ ΜΕ ΤΙΜΗ ΠΩΛΗΣΗΣ». Κάθε αντίτυπο που διατίθεται προς πώληση και δεν φέρει την παραπάνω ένδειξη θεωρείται κλεψίτυπο και ο παραβάτης διώκεται σύμφωνα με τις διατάξεις του άρθρου 7 του νόμου 1129 της 15/21 Μαρτίου 1946 (ΦΕΚ 1946,108, Α').